Niveau(x) d'enseignement: 7 Numéro EDUQ : 8570880
Centre d'intérêt: 774

Fournier, Claire.

Conception informatique d'un micromonde en science sociales : rapport de 1984/1985 / Claire Fournier et Jacques Lafeuille ; [pour le] Cégep de Chicoutimi. -- Chicoutimi : Le Collège, 1985.
92, [7] f. : 42 graph. ; 28 cm.

Résumé au début du v. Bibliogr. : f. 92.

Descripteur(s): *Concept *Didacticiel *Informatique *Sciences sociales *Collégial (Niveau)

Identificateur(s) de sujet: *Micro-monde *Checkland (Méthode)

Le rapport fait état de la recherche menée au cours de l'année 1984-1985 sur la conception informatique d'un micromonde en sciences sociales. Il s'agit de la deuxième étape d'une recherche plus vaste visant la mise au point d'un didacticiel en sciences sociales. Le rapport présente les cinq termes centraux assimilables aux structures-mères des sciences sociales (association, code, échange, forme, influence) sous leur forme graphique, de même que les différentes raisons qui ont amené les auteurs à choisir ces termes. Dans une perspective de devis informatique, chacune des représentations graphiques est décrite d'abord sous forme de page-écran et ensuite sous forme de tables de décision.

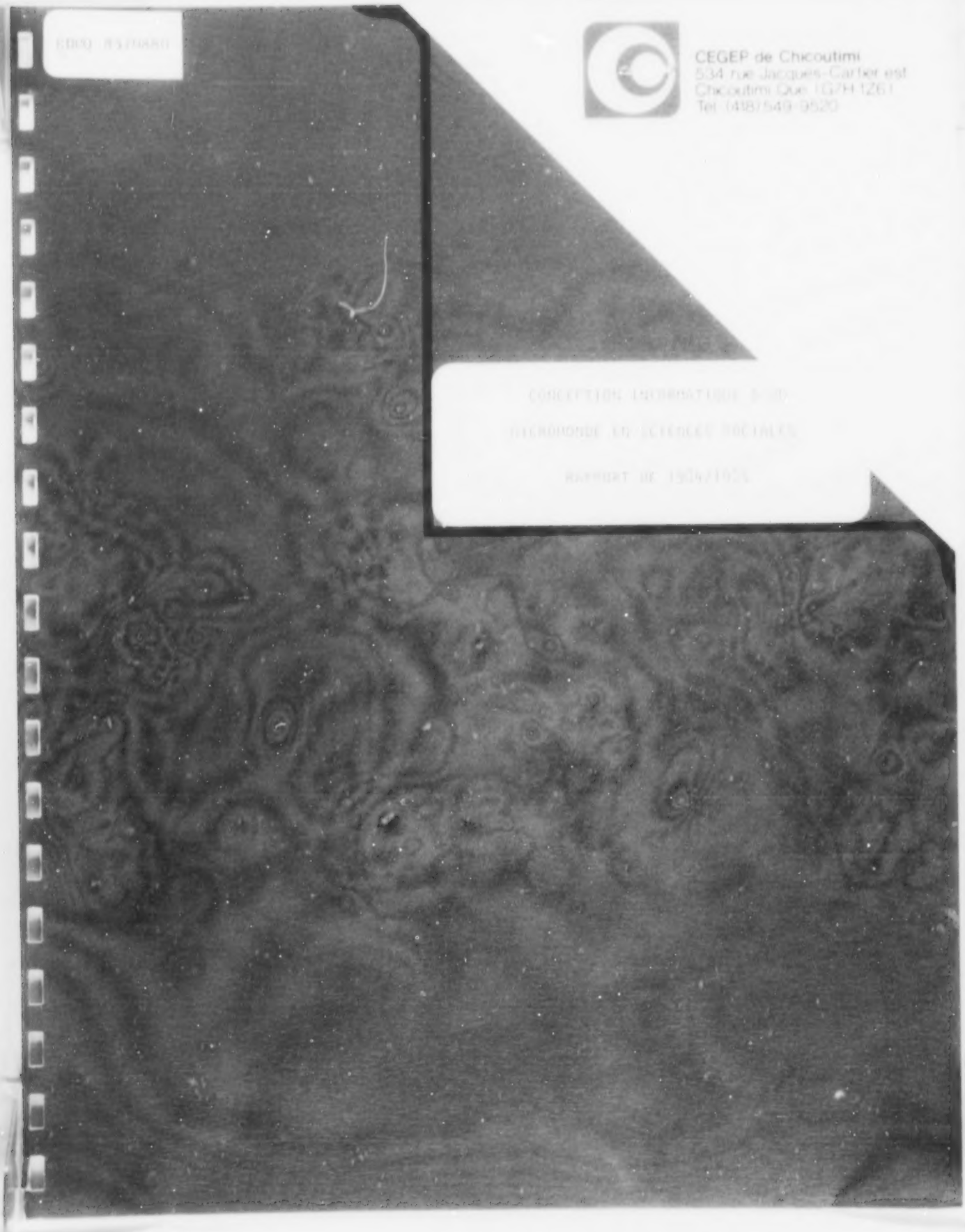
La première partie du rapport rappelle le cheminement global de la recherche dans le cadre de la méthodologie de Checkland, de même que la parenté qui existe entre le micromonde des sciences sociales et celui de la géométrie-tortue du langage Logo. La deuxième partie présente le passage de la conceptualisation du micromonde à la conception informatique. On y voit comment les termes centraux des sciences sociales sont transformés en représentations graphiques sur un écran cathodique. A cela s'ajoutent les caractéristiques du langage permettant de programmer les représentations graphiques. La troisième partie du rapport décrit de façon technique le langage animant le micromonde des sciences sociales.

Le micromonde de sciences sociales est suffisamment élaboré pour passer à l'étape de la réalisation informatique. Le concept de micromonde (dont la tortue-logo est une application) peut être appliqué à des domaines comme ceux des sciences humaines. Cela est d'autant plus intéressant et important que ces domaines d'application sont négligés par les applications pédagogiques de l'ordinateur.

Type de document: r
Notice(s) connexe(s): 8470952
Financement: Bénéficiaire - Collège de Chicoutimi
Organisme(s) source(s) - MESST-DGEC
Programme(s) de subvention - PROSIP

EBSQ 8320880

CEGEP de Chicoutimi
534 rue Jacques-Cartier est
Chicoutimi Que. G7H 1Z6
Tel. (418) 549-9520



EDUQ 8570880

Conception informatique d'un micromonde en sciences sociales

RESUME DU

RAPPORT DE 1984/1985

Claire Fournier et Jacques Lafeuille

Cegep de Chicoutimi

juin 1985

RESUME

La conception informatique d'un micromonde de sciences sociales a permis de concrétiser sur micro-ordinateurs des concepts abstraits définis lors de la phase précédente de la recherche. En effet, ce modèle théorique cerné par la conceptualisation du micromonde de sciences sociales avait fait ressortir que la spécificité d'un micromonde reposait sur le lien étroit entre les commandes d'un langage et les structures-mères d'une discipline. Pour les sciences sociales, nous avons tenté de définir les structures-mères des quatre disciplines enseignées au Cegep. Les résultats nous ont donné cinq termes centraux assimilables aux structures-mères des sciences sociales, à savoir association pour sciences sociales; code pour anthropologie; échange pour économique et influence pour politique.

Ce rapport présente les cinq termes centraux sous leur forme graphique, de même que les différentes raisons qui nous ont amenés à faire les choix que nous avons faits. Dans une perspective de devis informatique, chacune des représentations graphiques choisies y est décrite d'abord sous forme de pages-écran et ensuite sous forme de tables de décision. Ces dernières donnent, sous forme de tableaux, les règles devant régir chacune des actions graphiques choisies.

Vous pouvez avoir une idée des caractéristiques graphiques retenues pour chacun des termes centraux identifiés par le tableau suivant:

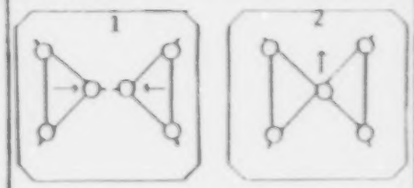
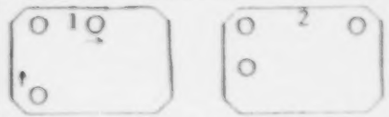
Caractéristiques graphiques / Termes centraux	REPRESENTATIONS GRAPHIQUES	COMPORTEMENTS GRAPHIQUES	ILLUSTRATION
Association	Le micromonde permet de programmer uniquement des relations.	Un élément doit toujours être en relation avec un autre. Seul, il disparaît.	
Code	La couleur attribue un code à l'élément.	Deux éléments pourront être en relation s'ils ont des couleurs composées d'au moins une couleur commune.	Violet et Vert communiquent par le Bleu
Echange	L'échange est symbolisé par des transferts de taille.	Un élément peut donner ou recevoir des portions de taille.	
Forme	Une forme géométrique groupe un ensemble d'éléments et de relations.	Membre d'une forme, un élément possède des antennes. La forme se déplace sur l'écran selon une direction ou une vitesse. Elle peut rejeter ou se juxtaposer à d'autres formes.	
Influence	Le déplacement d'un élément par rapport à un autre.	Un élément peut repousser un autre élément ou l'attirer jusqu'à l'absorption.	

Tableau A: Résumé des caractéristiques graphiques des termes centraux

Conception informatique d'un micromonde en sciences sociales

RAPPORT DE 1984/1985

Claire Fournier et Jacques Lafeuille

Cegep de Chicoutimi

Juin 1985

REMERCIEMENTS

Nous voulons remercier les personnes qui nous ont aidés d'une manière ou d'une autre à la réalisation de la conception informatique d'un micromonde en sciences sociales.

Merci à:

- Pierre Fraser qui nous a donné des conseils méthodologiques et un feedback lors de nos présentations;

- Jean-Claude Hudon qui a solutionné avec empressement nos problèmes administratifs;

- Mario Carrière, de la Firme Les Systèmes d'ordinateur Logo Inc., qui a commencé à travailler à un prototype du micromonde;

- Eric Laroche, de la Firme Debugg Informatique Inc., qui a fait l'étude de faisabilité;

- aux membres du département qui nous ont manifesté de l'intérêt sous diverses formes;

- aux animateurs du CRAPO qui ont dû partager avec nous l'utilisation des ordinateurs.

TABLE DES MATIERES

INTRODUCTION ..... page 6

I- RAPPEL METHODOLOGIQUE ET OBJET DE LA RECHERCHE 7

1. Rappel de la methodologie 8

2. La methodologie et le sous-système de conception informatique 15

II- LA STRUCTURE DU MICROMONDE 27

1. Les objectifs du langage 30

2. Le mode de representation des termes centraux 37

3. Les représentations des termes centraux 42

4. Les représentations les unes par rapport aux autres 53

III- LA REALISATION DU SOUS SYSTEME DE SPECIFICATION INFORMATIQUE 56

1. Le structurogramme général 58

2. Les pages-écran 61

3. Table de décision 80

CONCLUSION 90

REFERENCES 92

ANNEXE: Document d'orientation sur le projet de micromonde en sciences sociales: Analyse de faisabilité

LISTE DES FIGURES

Figure 1: L'approche des systèmes souples de Checkland appliquée à la recherche 10

Figure 2: Modèle conceptuel du système global de la recherche 12

Figure 3: Pertinence du modèle avec le domaine des sciences sociales 18

Figure 4: Résumé des résultats 20

Figure 5: Sous-système de conception informatique du micromonde 25

Figure 6: L'approche des systèmes souples de Checkland appliquée à la conception informatique 26

Figure 7: Sous-système d'opérationnalisation de la structure du micromonde 29

Figure 8: Sous-système de spécification informatique 57

Figure 9: Structurogramme general 59

Figure 10: Aperçu du contenu de la section 212.00 du structurogramme général 60

Figures 11 à 42 : Représentations graphiques ..... 62 à 79

LISTE DES TABLEAUX

Tableau 1: Interférence des modes de représentations par rapport aux règles 40

Tableau 2: Interférence des relations les unes par rapport aux autres 54

LISTE DES TABLES DE DECISIONS

Table 1: Association-échange 81

Table 2: Erreurs de la primitive échange 82

Table 3: Association-influence 83

Table 4: Erreurs de la primitive influence 84

Table 5: Association-code 85

Table 6: Erreurs de la primitive code 86

Table 7: Association-forme 87

Table 8: Erreurs de la primitive groupe 88

Table 9: Fonction contrôle 89

RAPPORT 1984-1985

Introduction

Le présent rapport fait état de la recherche menée au cours de l'année 1984-1985 sur la conception informatique d'un micromonde en sciences sociales. Il s'agit de la deuxième étape d'une recherche plus vaste visant la mise au point d'un didacticiel en sciences sociales.

Nous avons jugé important de bien situer le lecteur par rapport à l'ensemble de la recherche. C'est pourquoi la première partie rappelle le cheminement global de la recherche dans le cadre de la méthodologie de Checkland, de même que la parenté qui existe entre le micromonde des sciences sociales et celui de la géométrie-tortue du langage Logo.

La seconde partie du rapport présente le passage de la conceptualisation du micromonde à la conception informatique. Nous y verrons comment les termes centraux des sciences sociales sont transformés en représentations graphiques sur un écran cathodique. A cela s'ajoute les caractéristiques du langage permettant de programmer les représentations graphiques.

La troisième partie du rapport décrit de façon technique le langage animant le micromonde des sciences sociales.

I- RAPPEL METHODOLOGIQUE ET OBJET DE LA RECHERCHE

1. RAPPEL DE LA METHODOLOGIE

   1.1 Le sous-système 1

   1.2 Le sous-système 2

   1.3 Le sous-système 3

   1.4 Le sous-système 4

2. LA METHODOLOGIE ET LE SOUS-SYSTEME DE CONCEPTION INFORMATIQUE

   2.1 La problématique

   2.2 L'ancrage

   2.3 Le modèle du sous-système de conception informatique

Première partie

Rappel méthodologique et objet de la recherche

En juin 1984, nous avons rédigé un premier rapport de recherche présentant le modèle conceptuel du micromonde, l'état de la question de même que les concepts de sciences sociales sur lesquels repose le micromonde en construction. Le tout a été articulé par une démarche systemique: la méthodologie de Checkland.

Cette méthodologie a présenté, dès le début de la recherche, l'immense avantage de donner un aperçu global de la recherche en termes de tâches à réaliser, sans en scléroser la démarche, ni en exclure les diverses méthodes analytiques utiles selon les différentes phases des travaux. En effet, la méthodologie de Checkland nous aide, à chaque étape, à poser un diagnostic sur la situation présente et à intervenir en conséquence par une mise à jour de la problématique et du point de vue adopté. Pour bien comprendre ce que cela signifie, voici un bref rappel de l'application que nous avons fait de la méthodologie de Checkland.

1. Rappel de la méthodologie

L'approche de Checkland nous a fourni un instrument pour appréhender nos problèmes de structuration, d'organisation, de méthodologie et pour

développer un modèle d'intervention visant le bon fonctionnement de l'entreprise qu'est notre recherche. Nous vous présentons notre essai d'application en suivant les six étapes suivantes:

1. La problématique de la situation telle qu'elle se présentait au début de la recherche;

2. l'ancrage représentant le système global de la recherche;

3. le modèle conceptuel tiré de l'ancrage;

4. le sous-système de conceptualisation du micromonde;

5. la comparaison du modèle conceptuel et la problématique;

6. l'application du sous-système de conceptualisation du micromonde.

Ces différentes étapes appliquées à notre recherche sont illustrées dans le schéma suivant:

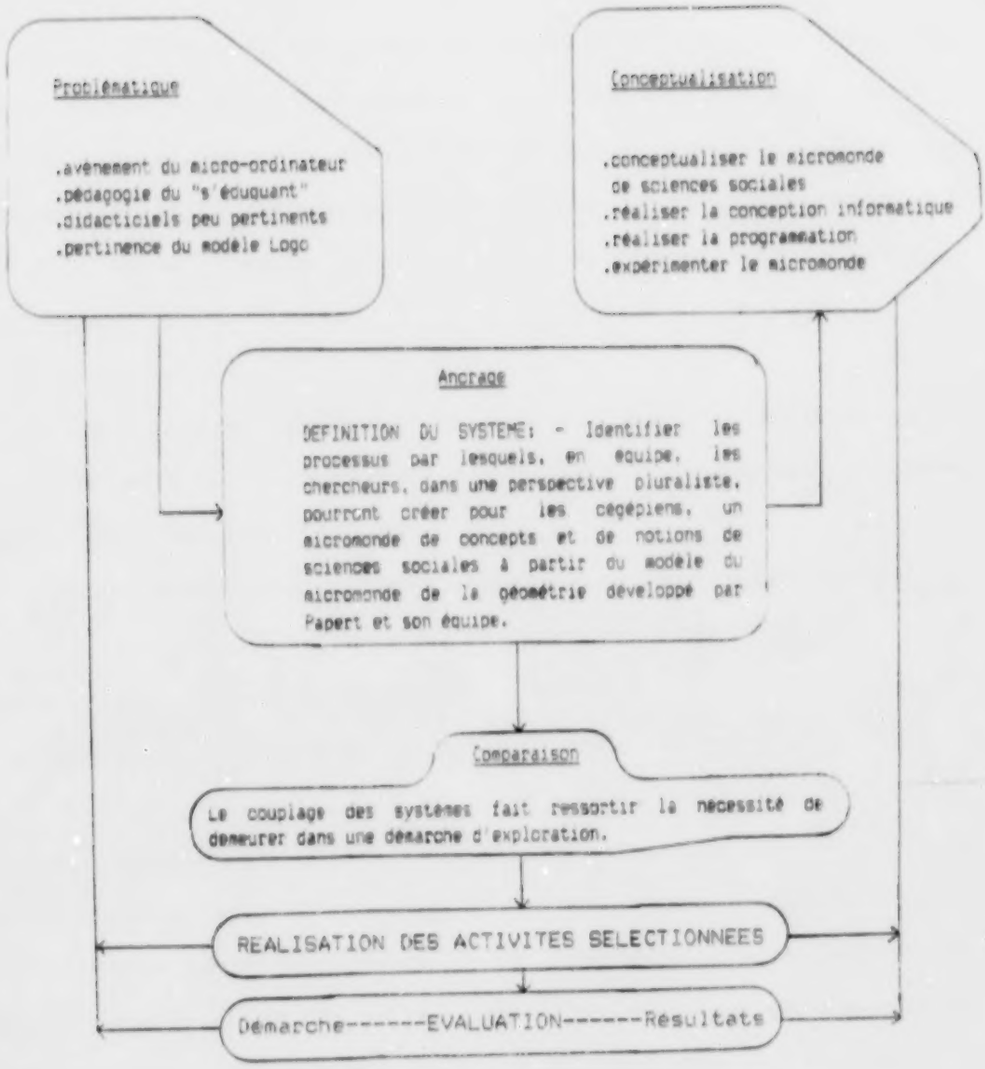

Problématique
.avènement du micro-ordinateur
.pédagogie du "s'éduquant"
.didacticiels peu pertinents
.pertinence du modèle Logo
Conceptualisation
.conceptualiser le micromonde de sciences sociales
.réaliser la conception informatique
.réaliser la programmation
.expérimenter le micromonde
Ancrage
DEFINITION DU SYSTEME: - Identifier les processus par lesquels, en équipe, les chercheurs, dans une perspective pluraliste, pourront créer pour les cegépiens, un micromonde de concepts et de notions de sciences sociales à partir du modèle du micromonde de la géométrie développé par Papert et son équipe.
Comparaison
Le couplage des systèmes fait ressortir la nécessité de demeurer dans une démarche d'exploration.
REALISATION DES ACTIVITES SELECTIONNEES
Démarche------EVALUATION------Résultats

Figure 1: L'approche des systèmes souples de Checkland appliquée à la recherche ^

La vérification des hypothèses de la recherche implique un processus qui se déroulera sur plusieurs années académiques et qui sera supporté par des méthodes propres à chaque activité majeure. Pour s'assurer que le modèle rassemble des actions humaines, Checkland propose d'exprimer les activités du système par une liste de verbes reliés par flux appropriés.

La création d'un micromonde en sciences sociales suppose quatre activités majeures correspondant, elles-mêmes, à quatre systèmes d'activités. Ces activités sont:

1- conceptualiser le micromonde de sciences sociales (1983/1984);

2- faire la conception informatique du micromonde (1984/1985);

3- faire la réalisation informatique;

4- expérimenter le micromonde (à partir de sept. 1985).

Ces activites sont interreliées de façon à toujours pouvoir reviser le travail déjà fait. Il ne s'agit pas d'un travail linéaire où les définitions et les choix effectués à la première activité déterminent obligatoirement le reste de la recherche. Un processus de rétroaction assure une revision permanente des étapes de recherche en fonction de l'évaluation de l'évolution des connaissances et de la perception de la problématique. Décomposées en un modèle conceptuel de niveau plus restreint, ces quatre activités forment des sous-systèmes représentés par la figure suivante:

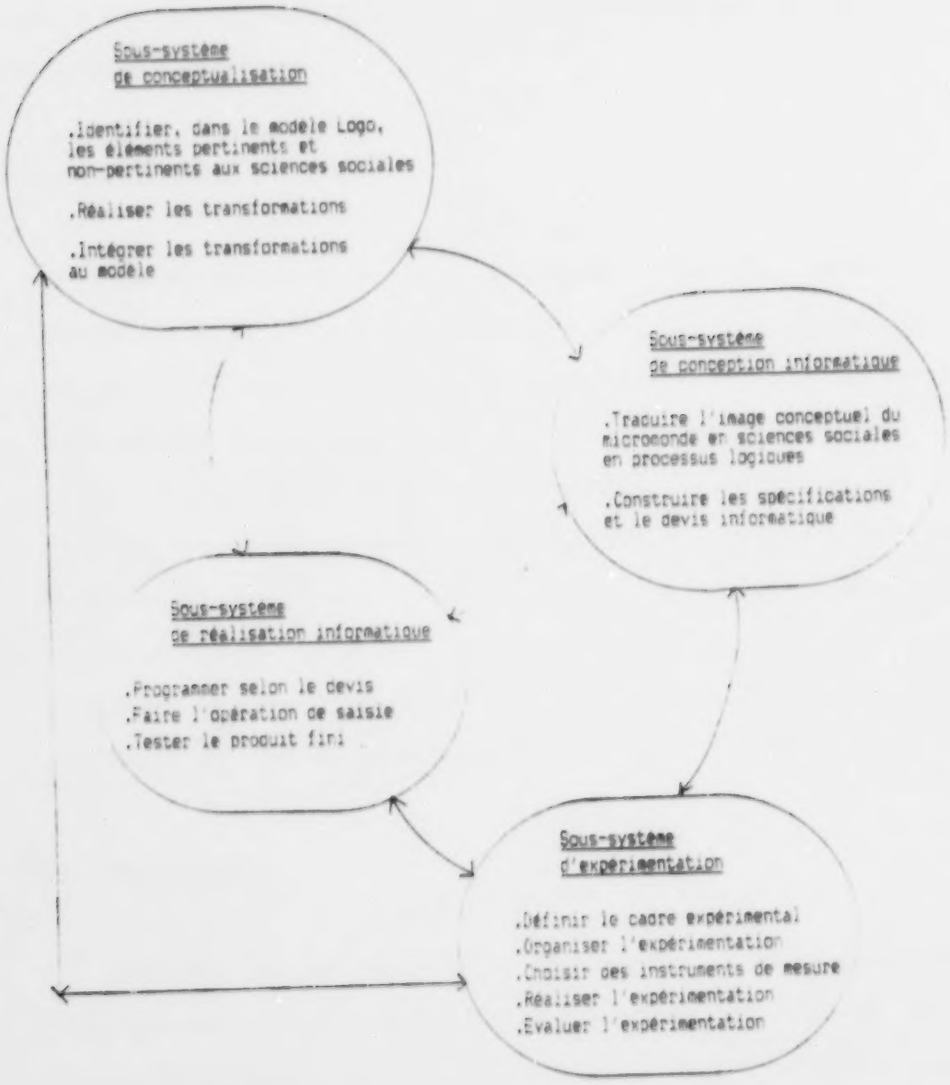

Sous-système de conceptualisation
.Identifier, dans le modèle Logo, les éléments pertinents et non-pertinents aux sciences sociales
.Réaliser les transformations
.Intégrer les transformations au modèle
Sous-système de conception informatique
.Traduire l'image conceptuel du microonde en sciences sociales en processus logiques
.Construire les spécifications et le devis informatique
Sous-système de réalisation informatique
.Programmer selon le devis
.Faire l'opération de saisie
.Tester le produit fini
Sous-système d'expérimentation
.Définir le cadre expérimental
.Organiser l'expérimentation
.Choisir des instruments de mesure
.Réaliser l'expérimentation
.Evaluer l'expérimentation

Figure 2: Modèle conceptuel du système global de la recherche[3]

Comme on peut le constater, chaque sous-système englobe un processus de recherche qui sera défini par des étapes méthodologiques. En effet, chacun a des objectifs spécifiques et une hypothèse à vérifier selon une méthodologie pertinente.

1.1 Le sous-système 1:

La conceptualisation du micromonde vise à cerner les caractéristiques d'un environnement propice à l'utilisation d'enseignement assisté par ordinateur en sciences sociales au niveau collégial. Il s'agit de définir, pour les sciences sociales, les notions, les structures-mères et les procédés d'apprentissage en vue de créer "un micromonde autonome et délimité"[4]. Cet objectif nous amène à prendre comme hypothèse qu'il est possible de rendre opérationnel les principes et les critères de conception du micromonde de façon à identifier les composantes spécifiques d'un micromonde en sciences sociales.

1.2 Le sous-système 2:

La conception informatique a pour objectif de réaliser sur micro-ordinateur la transformation de l'objet de transition de façon à créer le micromonde en sciences sociales. Il s'agit de traduire les concepts sous forme de processus informatique. Cet objectif amène

l'hypothese que les notions et les structures-mères, une fois définies sous forme de processus, pourront être représentées par un objet informatisé conforme aux caractéristiques cernées par le sous-système 1 de la recherche.

1.3 Le sous-système 3:

La réalisation informatique implique le choix d'un langage informatique et d'un appareil pouvant supporter les exigences du devis, la programmation, l'opération de saisie du programme, la vérification et l'évaluation du programme. Ce sous-système poursuit la vérification de l'hypothèse du système précédent.

1.4 Le sous-système 4:

L'experimentation du micromonde vise à vérifier l'atteinte des objectifs pédagogiques du didacticiel. Ces objectifs ont été définis par le sous-système 1 et présentés dans le rapport de recherche 1983/1984 [5].

L'experimentation pedagogique déclenche le processus d'évaluation, correction et enrichissement du micromonde.

2. La méthodologie et le sous-système de conception informatique

La réalisation, au cours de l'année 83-84, des activités du sous-système de conceptualisation nous a donné les éléments nécessaires pour réaliser les activités prévues au sous-système de conception informatique. La méthodologie de Checkland qui a été appliquée pour la définition globale de la recherche peut aussi servir à définir un modèle d'activités à un niveau plus restreint et plus spécifique de la recherche comme celui de la conception informatique. Pour clarifier l'ensemble des tâches pertinentes au sous-système de conception informatique, nous suivrons à nouveau les étapes de la méthodologie de Checkland décrites précédemment.

2.1 La problématique

La problématique se définit par une étude de l'état de la situation. En septembre 1984, la situation de la recherche pourrait se résumer ainsi: des ressources humaines et matérielles sont disponibles pour poursuivre une recherche reposant sur la conceptualisation d'un micromonde en sciences sociales.

a) Les ressources disponibles

Suite à la présentation du projet de recherche au programme de

subvention à l'innovation pédagogique, le collège a obtenu, du ministère de l'éducation, les sommes nécessaires pour libérer deux professeurs de leur tâche d'enseignement et assurer à ces derniers un petit budget de fonctionnement. Cette subvention et la collaboration du collège ont assuré les ressources humaines et matérielles pour réaliser le projet.

b) La conceptualisation du micromonde de sciences sociales

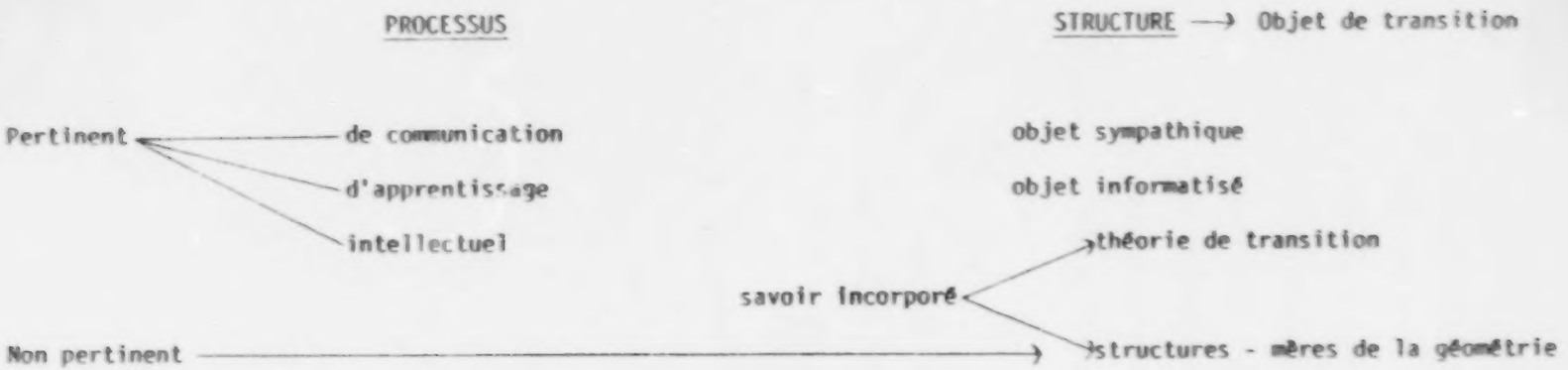
La recherche de l'an dernier a permis d'identifier un modèle général de micromonde en distinguant les éléments généralisables à plusieurs disciplines de ceux qui sont spécifiques à une discipline.

L'identification des éléments pertinents et non-pertinents a fait ressortir le modèle d'apprentissage Logo en termes de processus et de structure. Les processus sont dominés par le fait que l'utilisateur programme l'ordinateur pour enseigner des "choses" à l'écran. L'objet de transition qu'est la tortue comprend des procédures simples enseignées avec des mots de tous les jours. Elle est la structure du micromonde qui permet l'actualisation des processus. Tous les éléments du micromonde de la géométrie, à l'exception d'un seul, sont pertinents avec un micromonde de sciences sociales. L'élément non-pertinent est

structurel parce qu'il fait partie de l'objet de transition qu'est la tortue. Il s'agit des primitives utilisées pour commander la tortue. Ces primitives ont pour caractéristiques d'exprimer les structures-meres spécifiques à la géométrie, ce qui ne peut pas s'appliquer aux sciences sociales.

Ainsi nous pouvons conserver, pour le micromonde de sciences sociales, tous les processus identifiés de même que les aspects sympathique et informatique de la structure. Il nous reste à modifier les primitives pour qu'elles expriment les structures-mères (structures mentales) des sciences sociales plutôt que celles de la geometrie.

La figure suivante résume le problème posé par la recherche en général.



LE PROBLEME A RESOUDRE

MICROMONDE	⟶	STRUCTURES-MERES	⟶	PRIMITIVES
Tortue Logo	⟶	orientation direction	⟶	avance, recule gauche, droite
Sciences Sociales	⟶	?	—	?

Figure 3: Pertinence du modèle avec le domaine des sciences sociales

Les travaux de recherche de l'année dernière ont été dominés par l'identification des structures-mères des sciences sociales. Pour les trouver, nous avons utilisé la théorie des clusters[1], qui nous a permis d'identifier des "core terms" qui, de par leurs caractéristiques, sont traduisibles en structures-mères, puis en primitives d'un langage. Les "core terms" sont des termes nécessaires pour référer à un domaine de connaissance mais insuffisants pour définir ce domaine. Une fois les termes centraux des sciences sociales identifiés, nous les avons validés au moyen d'une enquête Delphi[7].

Les résultats de l'enquête ont fait l'objet d'une autre validation par les chercheurs, ce qui a permis de dégager deux postulats et cinq termes centraux.

POSTULATS: 1. Le caractère humain de tout élément des définitions;
2. L'approche par processus.

TERMES CENTRAUX	DEFINITIONS
association:	agents ou unités qui vivent les phénomènes politiques, économiques, culturels, et sociaux.
code	système d'éléments mentaux significatifs pour les agents et les unités.
échange	le passage d'un agent (ou d'une unité) à un autre agent (ou une autre unité) de valeurs plus ou ou moins équivalentes.
force	les agencements des agents et des unités dans un tout.
influence	action qu'exerce un agent ou une unité sur un autre agent ou une autre unité. Cela peut se produire dans un contexte décisionnel (autorité, pouvoir) ou dans un contexte non-décisionnel (négociation, tradition).

Figure 4: Résumé des résultats

Une fois ces termes centraux définis, il faut les traduire en primitives d'un langage qui servira à commander l'objet de transition du micromonde de sciences sociales. C'est le défi que posait la situation de notre recherche en septembre dernier.

2.2 L'ancrage

L'ancrage sert à déterminer la perspective sous laquelle la problématique sera étudiée. Elle implique trois (3) étapes:

1- choix du système pertinent au problème étudié:

2- définition du système:

3- validation de la définition.

1- Choix du système pertinent au problème étudié:

Le choix du système pertinent se fait à partir de la question problématique, et de notre point de vue sur cette question. Pour nous, il s'agit de cerner le système global de la recherche.

a) La question problématique

La situation problématique se trouve au niveau de la transformation des termes centraux des sciences sociales en primitives d'un langage informatique. Cette situation problématique pourrait se traduire par la question suivante: comment faire pour qu'un ordinateur puisse présenter aux cégépiens des concepts de sciences sociales de façon à ce que ces concepts soient, pour lui, des outils de programmation? Pour répondre à cette question, nous avons fait un certain

nombre de choix qui illustrent notre point de vue.

b) Notre point de vue:

Pour être des outils de programmation, les concepts doivent être partie intégrante d'un langage informatique. C'est pourquoi nous parlons de primitives d'un langage. Toutefois, il ne nous apparait pas nécessaire d'avoir toutes les fonctions d'un langage informatique mais uniquement celles qui permettent la manipulation la plus versatile possible des concepts. Pour cela, le fonctionnement du micromonde a besoin d'un langage favorisant la programmation par procédure et qui opère avec les fonctions logiques et mathématiques.

Comme les chercheurs ne sont pas des informaticiens, ils perçoivent leurs rôles comme pédagogues des sciences sociales. Aussi, leur tâche consistera à faire une description de la performance qu'ils attendent du logiciel. Cette description aura ceci de particulier qu'elle se rapprochera suffisamment du fonctionnement de l'ordinateur pour qu'un programmeur puisse en faire aisément la codification. Une attention toute pédagogique sera accordée au système d'aide.

2- La définition du sous-système de conception informatique:

Notre point de vue nous permet de définir le sous-système d'activités que nous devons conceptualiser et appliquer par la suite. Voici les termes de la définition du sous-système:

> Le sous-système de la conception informatique identifie les activités par lesquelles les chercheurs, avec l'aide de consultants en micro-informatique, pourront décrire pour des programmeurs les termes centraux des sciences sociales transformés en primitives d'un langage informatique spécifique au micromonde des sciences sociales.

3- Validation de la définition du système:

Que vaut cette définition? Cette dernière est jugée valide parce que les chercheurs reconnaissent, premièrement, qu'elle s'inscrit dans la logique interne du projet et, deuxièmement, qu'elle informe sur les six (6) aspects fondamentaux d'un système d'activités humaines à savoir: le point de vue, les acteurs, les utilisateurs, les propriétaires ou preneurs de décision, l'environnement et les activités de transformation.

2.3 Modèle conceptuel du sous-système de conception informatique

Le modèle conceptuel du sous-système de conception informatique implique un processus qui se déroulera au cours de l'année académique 1984-1985. Pour s'assurer que le modèle rassemble des actions humaines, Checkland propose d'exprimer les activités du système par une liste de verbes reliés par flux appropriés.

La conception informatique du micromonde en sciences sociales suppose deux activités majeures correspondant elles-mêmes à deux (2) sous-systèmes d'activités. Ces activités sont:

1- transformer les termes centraux du micromonde de sciences sociales en primitives;

2- construire les spécifications informatiques.

Ces activités sont interreliées de façon à toujours pouvoir reviser le travail déjà fait.

Décomposées en un modèle conceptuel de niveau plus restreint, ces deux (2) activités forment ces sous-systèmes représentés par la figure suivante:

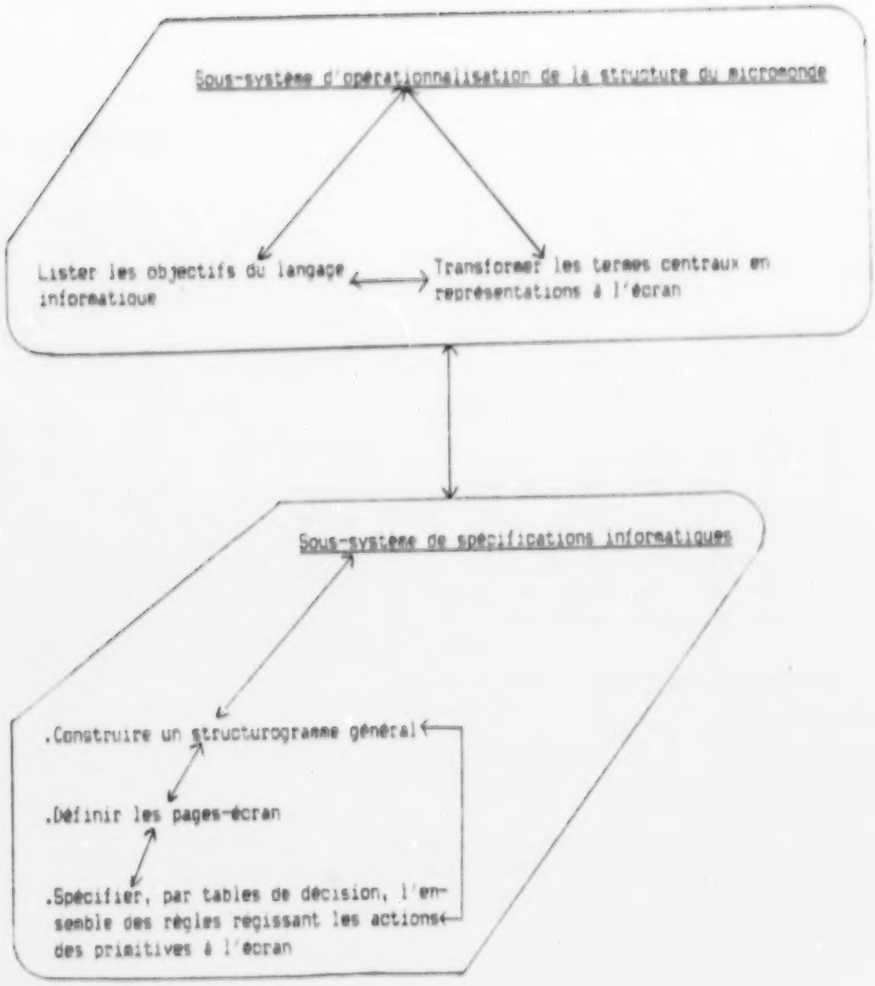

Sous-système d'opérationnalisation de la structure du micromonde
Lister les objectifs du langage informatique
Transformer les termes centraux en représentations à l'écran
Sous-système de spécifications informatiques
.Construire un structurogramme général
.Définir les pages-écran
.Spécifier, par tables de décision, l'ensemble des règles régissant les actions des primitives à l'écran

Figure 5: Sous-système de conception informatique du micromonde

Si on veut resumer la démarche de la deuxième étape de la recherche, voici un tableau-synthèse.

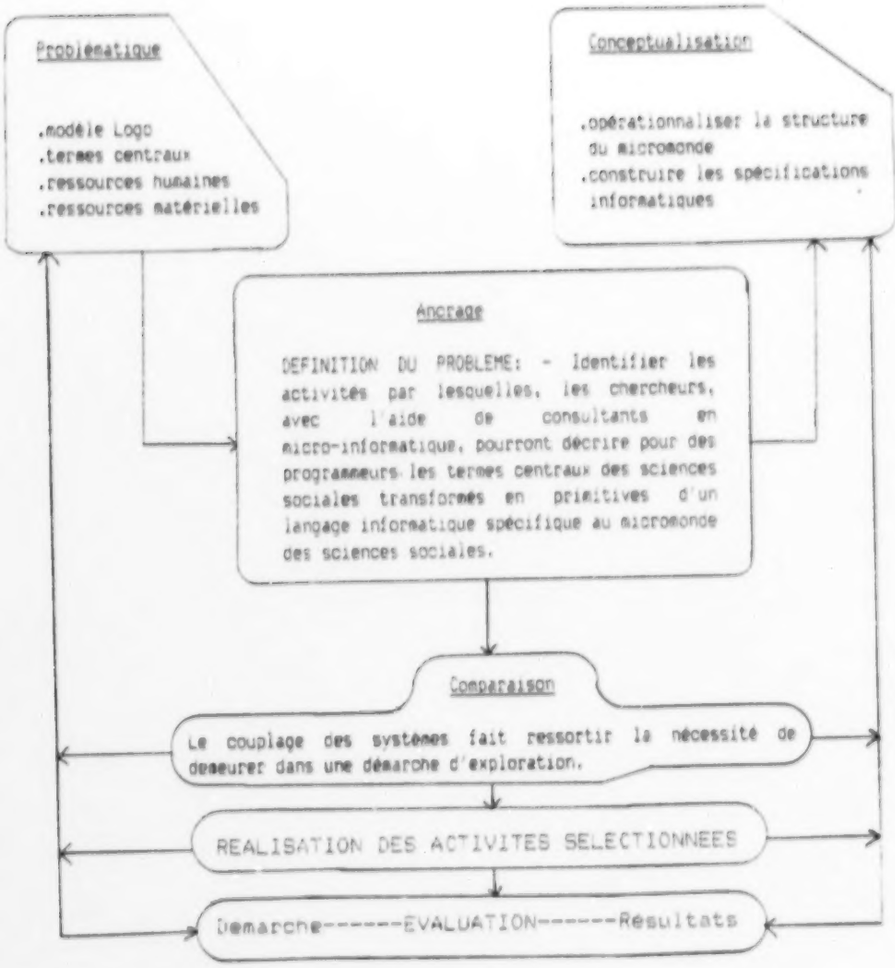

Problématique
.modèle Logo
.termes centraux
.ressources humaines
.ressources matérielles
Conceptualisation
.opérationnaliser la structure du micromonde
.construire les spécifications informatiques
Ancrage
DEFINITION DU PROBLEME: - Identifier les activités par lesquelles, les chercheurs, avec l'aide de consultants en micro-informatique, pourront décrire pour des programmeurs les termes centraux des sciences sociales transformés en primitives d'un langage informatique spécifique au micromonde des sciences sociales.
Comparaison
Le couplage des systèmes fait ressortir la nécessité de oeuvrer dans une démarche d'exploration.
REALISATION DES ACTIVITES SELECTIONNEES
Demarche------EVALUATION------Resultats

Figure 6: L approche des systèmes couplés de Checkland appliquée à la conception informatique

II- LA STRUCTURE DU MICROMONDE

1. LES OBJECTIFS DU LANGAGE

   1.1 Les objectifs généraux

   1.2 Les objectifs opérationnels

2. LE MODE DE REPRESENTATION DES TERMES CENTRAUX

   2.1 Les règles

   2.2 Les capacités des ordinateurs actuels

   2.3 Le mode de représentation

3. LES REPRESENTATIONS DES TERMES CENTRAUX

   Echange (résumé)

   Code (résumé)

   Influence (résumé)

   Forme-agencement (resumé)

   Forme-groupement (résumé)

4. LES REPRESENTATIONS LES UNES PAR RAPPORT AUX AUTRES

Deuxième partie

La réalisation du sous-système de transformation des termes centraux en primitives

Maintenant que nous sommes situés méthodologiquement et par rapport aux résultats obtenus de l'étape de la conceptualisation, voyons en quoi consiste le sous-système d'opérationalisation de la structure du micromonde.

Le sous-système d'opérationalisation de la structure du micromonde comporte les activités de la figure 7. Les activités principales, soit lister les objectifs du langage informatique et transformer les termes centraux en représentations à l'écran, sont définies par des activités plus simples.

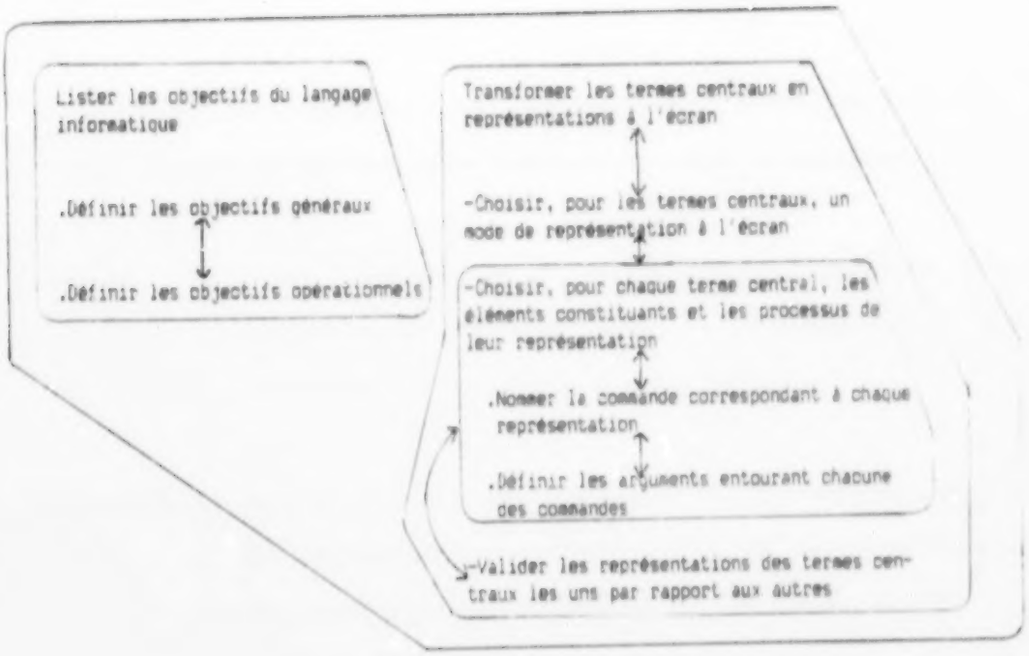

Lister les objectifs du langage informatique
.Définir les objectifs généraux
.Définir les objectifs opérationnels
Transformer les termes centraux en représentations à l'écran
-Choisir, pour les termes centraux, un mode de représentation à l'écran
-Choisir, pour chaque terme central, les éléments constituants et les processus de leur représentation
.Nommer la commande correspondant à chaque représentation
.Définir les arguments entourant chacune des commandes
-Valider les représentations des termes centraux les uns par rapport aux autres

Figure 7: Sous-système d'opérationnalisation de la structure du micromonde

Les resultats de la realisation de ce sous-système seront presentes sous les titres suivants: les objectifs du langage, le mode de représentation des termes centraux, les représentations des termes centraux, les représentations les unes par rapport aux autres.

1. Les objectifs du langage

1.1 Objectifs généraux

Le langage doit permettre, par la manipulation de ses primitives, l'apprentissage de notions et de concepts de sciences sociales.

Langage : C'est l'ensemble de toutes les phrases possibles par l'utilisation de fonctions, de primitives et de procedures particulières à l'utilisateur.

Manipulation: Afin de réaliser des projets, l'utilisateur construit des procédures (programmes) à l'aide des fonctions, des primitives des procédures déjà définies ou utilise directement les primitives du langage.

Primitives : Ce sont les commandes initiales du langage auxquelles s'additionnennt les procédures construites par l'utilisateur.

1.2 Objectifs opérationnels

Le langage doit permettre à l'utilisateur:

1. De réaliser des projets en écrivant à l'aide des primitives des programmes ou procédures

1.1 Donner un nom à la procédure

1.2 Utiliser selon les projets les primitives du terme central code

1.3 Utiliser selon les projets les primitives du terme central echange

1.4 Utiliser selon les projets les primitives du terme central forme

1.5 Utiliser selon les projets les primitives du terme central influence

1.6 Utiliser des procédures déjà définies

1.7 Utiliser selon les projets les autres capacités du langage

- Répéter:

  tant que

  répète...jusque

- Opérer de façon conditionnelle:

  si...et

si...ou

si...alors

si...alors...sinon

si...tout vrai

teste...si vrai

teste...si faux

- Opérer en utilisant des variables
- Opérer de façon mathématique:

additionner

soustraire

multiplier

diviser

plus grand que

plus petit que

égal

- Contrôler le déroulement:

arrêter la procédure de façon définitive

arrêter la procédure de façon temporaire

exécuter la procédure

- Affecter l'écran ou l'objet de transition:

effacer l'écran

positionner l'objet de transition à son point de départ

positionner l'objet de transition dans l'écran

faire disparaître ou apparaître l'objet de transition s'il y a lieu

retourner les coordonnées de l'objet de transition

donner l'orientation de l'objet de transition

retourner l'objet de transition à sa position par défaut

modifier la couleur de fond

écrire à l'imprimante

écrire à l'ecran

définir un écran circulaire

définir un écran fermé

1.8 Valider la procédure

2. De voir les résultats de l'utilisation de procédures

2.1 Appeler la procédure

- Vérifier le lexique
- Vérifier la syntaxe
- Representer à l'écran les comportements appelés dans la procédure

3. De stocker les nouvelles procédures du langage.

4. De réaliser des projets sans construire une procedure

4.1 Utiliser les primitives et les procedures déjà définies

- Utiliser selon les projets les primitives du terme central

code

- Utiliser selon les projets les primitives du terme central échange
- Utiliser selon les projets les primitives du terme central forme
- Utiliser selon les projets les primitives du terme central influence
- Utiliser selon les projets les procédures déjà définies

4.2 Voir les résultats de l'utilisation des primitives et/ou des procédures:

- Valider la phrase
- Contrôler la phrase:
  . Contrôler le lexique
  . Contrôler la syntaxe
  . Contrôler la logique
- Exécuter la phrase

5. De modifier une procédure déjà définie

5.1 Appeler la procédure

5.2 Modifier la procédure

6. De faire appel à un système d'aide

6.1 Aider au niveau de la syntaxe

6.2 Aider au niveau lexical

6.3 Aider au niveau des procédures:

- Sauvegarde des procédures
- Chargement des procédures
- Effacement de procédures
- Modification du nom d'une procédure
- Copie de procédures
- Impression des procédures

6.4 Aider au niveau de l'écran:

- Accès aux différents modes
- Les commandes d'édition
- Impression de l'écran

7. De manipuler les procédures déjà définies

7.1 Effacer une procédure

7.2 Renommer une procédure

7.3 Sauvegarder une procédure

7.4 Charger une procédure

7.5 Imprimer une procédure

8. De modifier les cadres (hors d'atteinte par programmation) du logiciel

9. De renvoyer des messages d'erreur adéquats

9.1 Contrôler le lexique

9.2 Contrôler la syntaxe

9.3 Contrôler la logique

9.4 Associer les messages d'erreur adéquats:

- Renvoyer au lexique

- Renvoyer à la syntaxe
- Renvoyer à la logique
- Renvoyer au système d'exploitation

2. Le mode de représentation des termes centraux

La représentation des termes centraux est constituée de ce qui apparaît à l'écran suite à la programmation des primitives découlant de ces termes centraux.

Ces représentations s'inscrivent dans un mode quelconque. Ce mode est en quelque sorte à l'intersection du savoir des utilisateurs et des possibilités des ordinateurs en regard des termes centraux retenus. Il s'agit donc pour nous de fixer les règles découlant de l'objectif de la manipulation des primitives (structures-mères) à l'intérieur du modèle du micromonde, d'inventorier les possibilités de représentation des ordinateurs et de choisir parmi ces possibilités celle qui enfreint le moins de règles.

2.1 Les regles

Ces règles découlent de l'objectif de la manipulation des primitives à l'intérieur du modèle du micromonde. Ces primitives découlent dans le modèle des structures-mères et donc pour nous, d'abord, des termes-centraux.

1. Les représentations des termes centraux doivent pouvoir s'intégrer au modèle du micromonde.

2. L'utilisateur accède aux termes-centraux (structures-mères) des sciences sociales par l'utilisation de primitives.

3. Les projets des utilisateurs ne doivent pas exiger une modification du savoir inclus dans les primitives.

4. Les projets éventuels doivent être motivants pour l'utilisateur.

5. La mise en route et la mise au point d'un projet doit exiger le minimum de connaissances.

6. L'utilisateur doit pouvoir réfléchir sur l'état de son projet à partir de résultats permanents.

7. Les résultats de la programmation à l'aide des primitives doivent

avoir un sens immédiat.

2.2 Les capacités des ordinateurs actuels

Les traitements qu'un ordinateur peut effectuer sont multiples. Mais les symboles disponibles pour représenter les résultats de ces traitements sont peu nombreux. En effet, l'ordinateur peut répondre par des symboles sonores ou visuels. Les symboles sonores vont du simple bruit à des musiques relativement complexes et les symboles visuels peuvent prendre la forme de mots, de nombres ou d'objets graphiques. Il est aussi possible d'obtenir toutes sortes de combinaisons de mots et/ou de nombres et/ou d'objets graphiques et/ou de sons.

2.3 Le mode de représentation

Dans le cadre des règles fixées à partir du modèle, nous pouvons voir dans quelle mesure les symboles disponibles pour représenter des résultats avec un ordinateur contreviennent à celles-ci. Le tableau 1 qui suit présente les résultats de cette analyse. Les règles sont indiquées par des numéros à l'horizontale et les modes de représentations disponibles sont indiqués à la verticale. Un (X) indique qu'un mode de représentation contrevient à une règle.

Règles / Symboles	1	2	3	4	5	6	7
Son	-	-	-	X	X	X	-
Mots	-	-	-	X	X	-	-
Nombres	-	-	-	X	X	-	-
Objets graphiques	-	-	-	-	-	-	-

Tableau 1: Interférence des modes de représentations par rapport aux règles.

On voit que les règles les plus cruciales sont celles portant sur les pré-requis exigés chez l'utilisateur et sur le caractère motivant des projets. Sans tomber dans l'analyse de l'opposition entre étudiants qui sont doués pour les mathématiques et ceux qui ne le sont pas, ou encore qui sont doues pour le français, on doit reconnaître qu'au niveau collegial, il s'agit d'une classification véhiculée par les étudiants eux-mêmes. C'est pourquoi les symboles des mots (la représentation serait alors assimilée à "du français") et des nombres (la représentation serait assimilée à des mathematiques) contreviennent pour nous aux règles 4 et 5.

La dimension du son pose aussi un problème dans la mesure où peu d'étudiants présentent suffisamment de connaissances ou d'intérêt pour ce domaine. Aussi, les symboles sonores contreviennent de façon plus critique encore à la règle 6. En effet, un agencement de sons

ne peut être permanent à l'oreille. L'analyse des résultats d'une programmation s'en trouverait alors compromise.

D'autre part, aucun symbole ne contrevient aux règles 1:2:3: et 7. En effet, en ce qui concerne la règle 1, le modèle du micromonde nous permet de penser que des primitives pourraient utiliser n'importe quel symbole pourvu que cela s'avère pertinent aux structures-mères à intégrer dans le micromonde.

La règle 2 indique que l'accès au savoir inclus dans les primitives, dans le cadre d'un projet qui doit être cohérent pour l'utilisateur, n'est possible que par l'utilisation des primitives. Le savoir inclus dans les primitives doit donc être complet. Il est concevable que cela soit réalisable quels que soient les symboles retenus.

De la règle 3, il découle que l'utilisateur ne doit pas être amené à vouloir attribuer à une ou des primitives d'autres symboles que ceux retenus par les concepteurs. Le mode de représentation doit être suffisamment riche pour y greffer un ensemble cohérent de primitives. A ce titre, il nous apparaît que, quel que soit le mode de représentation retenu, aucun symbole ne contreviendrait à cette règle.

Ainsi, de prime abord, seuls les objets graphiques ne

contreviennent pas à prime abord aux règles retenues. C'est donc dans le champ du dessin à l'aide d'objets graphiques que nous avons construit les représentations de nos termes centraux. Cela n'exclut pas que l'on utilise des mots, des nombres ou même le son pour compléter une représentation que le dessin ne pourrait pas couvrir.

3. Les représentations des termes centraux

3.1 Les caractéristiques des objets graphiques

Les caractéristiques utilisées sont reliées au mode de représentation choisi soit les objets graphiques. Il s'agit de répondre en fait à la question suivante: quelles caractéristiques des objets graphiques utiliserons-nous? Nous avons ainsi retenu le point comme élément de base. Ce point a une taille suffisante pour être visible. Il se situe à une distance pouvant varier de d'autres points. Aussi, ce point se voit attribuer une couleur selon les cas et la disposition des points les uns par rapport aux autres permet de créer des formes. Enfin, les points peuvent être reliés entre eux par des segments.

Dans le texte qui suit, le point sera nommé par l'"élément début" et l'"élément fin" ou tout simplement par l'"élément".

3.2 Echange

a) Choix de l'utilisateur:

A l'aide d'une commande d'échange et des paramètres qui la composent, l'utilisateur peut choisir, à partir de la position initiale d'un élément, l'endroit dans l'écran où situer l'élément fin de la relation (angle et longueur du segment); il peut choisir la taille des éléments en relation (la dotation des éléments) et les valeurs transférées d'un élément à l'autre (les flux).

---

commande d'échange, angle, longueur, dotation de l'élément début, dotation de l'élément fin, flux de l'élément début, flux de l'élément fin.

Une suite de commandes d'échange nous donne une suite de relations d'échange qui s'opèrent en vertu des paramètres spécifiés dans les commandes.

B) La dynamique:

1- Les éléments:

- La taille déterminée par des valeurs
- Un flux continu entre les éléments opérant dans les deux sens

2- Les principes:

- Le flux détermine la diminution, l'augmentation ou la stabilité de la taille des éléments.

3- Les processus:

- A toute augmentation de la taille d'un élément correspond la diminution de la taille d'un autre élément.
- Le flux détermine l'importance de la modification des volumes des éléments.
- Le flux entre les éléments est continuel, c'est-à-dire que les échanges se refont sans arrêt à l'intérieur des paramètres fixés par l'utilisateur jusqu'à épuisement de la taille des éléments.

4- Les cas limites:

- Un élément qui n'a pas de valeur, donc de taille, disparaît.
- Un élément sans interaction disparaît.

3.3 Code

a) Choix de l'utilisateur:

A l'aide de la commande d'utilisation de code et des paramètres qui la composent, l'utilisateur peut choisir, à partir de la position initiale d'un élément, l'endroit dans l'ecran où situer l'élément fin de la relation (angle et longueur du segment); il peut choisir les possibilités de communication des éléments en relation (les chances de communiquer des éléments) et le code (la couleur) de chaque élément.

---

commande d'utilisation de code, angle, longueur, chances de communiquer de l'élément début, chance de communiquer de l'élément fin, couleur de l'élément début, couleur de l'élément fin.

Une suite de commandes d'utilisation de code nous donne une suite de relations de communication qui s'opèrent en vertu des paramètres spécifiés dans les commandes.

b) La dynamique:

1- Les éléments:

- La composition des couleurs
- Les chances de communiquer

2- Les principes:

- Deux éléments peuvent communiquer si la composition de leur couleur comprend une couleur commune.
- Chaque élément utilise un nombre de chances de communiquer.

3- Les processus:

- Toute communication fait augmenter, chez l'élément qui la reçoit, ses chances de communiquer mais jamais au-delà de l'attribution par le programme.
- Toute communication fait diminuer, chez l'élément émetteur, ses chances de communiquer.
- Toute communication est continuelle, c'est-à-dire qu'elle se déroule à l'intérieur des paramètres fixés par l'utilisateur jusqu'à l'épuisement des chances de communiquer.

4- Les cas limites:

- Un élément qui utilise toutes ses chances de communiquer sans avoir reçu aucune communication disparaît.
- Un élément ne peut jamais voir augmenter son nombre de chances de communiquer au-delà du nombre de chances de communiquer attribuées par l'utilisateur.
- Un élément qui ne communique jamais et qui a reçu un nombre de communications égal à son quota X 2 + 1 voit ce quota diminué de un.

3.4 Influence

a) Choix de l'utilisateur:

A l'aide de la commande d'influence et des paramètres qui la composent, l'utilisateur peut choisir, à partir de la position initiale d'un élément, l'endroit dans l'écran où situer l'élément fin de la relation (angle et longueur du segment) et il peut choisir la force d'influence de chaque élément.

---

commande d'influence, angle, longueur, force d'attraction de l'élément début, force d'attraction de l'élément fin.

Une suite de commandes d'influence nous donne une suite de relations d'influence qui s'opèrent en vertu des paramètres spécifiés dans les commandes.

b) La dynamique:

1- Les éléments:

- Le déplacement des éléments
- Une force d'attraction par élément

2- Les principes:

- Tout élément a une force d'attration.
- Tout élément peut attirer un autre élément ou être attiré par un autre élément.

3- Les processus:

- Un élément ayant une force d'attraction inférieure à un autre élément est attiré par ce dernier.
- Deux éléments de force d'attraction égale ne se déplacent pas l'un vers l'autre.
- Un élément continuellement attiré par un autre élément est gobé par ce dernier.
- L'influence d'un élément joue continuellement sur tout élément avec lequel il est en relation.

4- Les cas limites:

- Un élément qui ne subit ni ne produit d'attraction disparaît.

3.5 Forme-agencement

a) Choix de l'utilisateur:

A l'aide de la commande d'agencement d'éléments et des paramètres qui la composent, l'utilisateur peut choisir le nombre de segments à utiliser successivement, l'angle qui sera utilisé pour créer l'agencement, la longueur à affecter aux segments, une vitesse à cet agencement et une direction à l'agencement dans l'écran.

---

commande d'agencement, nombre de segments, angle, longueur, vitesse de l'agencement, direction de l'agencement

---

b) La dynamique:

1- Les éléments:

- Des angles
- Des segments successifs
- La vitesse d'un agencement
- La direction d'un agencement

2- Les principes:

- Tout agencement est construit en traçant successivement des segments et des angles avec ces segments.
- Tout agencement a une vitesse et une direction.

3- Les processus:

- Tout agencement qui rencontre un autre agencement pivote ou ricoche sur l'autre agencement en fonction de sa vitesse et de sa direction.
- Le choc de deux agencements modifie la vitesse et la direction des agencements.

4- Les cas limites:

- Tout agencement qui n'a plus de vitesse disparaît.

3.6 Forme-groupement

a) Choix de l'utilisateur:

A l'aide de la commande d'agencement d'éléments et des paramètres qui la composent, l'utilisateur peut choisir le nombre de segments à utiliser successivement, l'angle qui sera utilisé pour créer l'agencement, la longueur à affecter aux segments, une vitesse à cet agencement et une direction à l'agencement dans l'écran.

---

commande d'agencement, nombre de segments, angle, longueur, vitesse de l'agencement, direction de l'agencement

---

b) La dynamique:

1- Les éléments:

- Des angles
- Des segments successifs
- La vitesse d'un groupement
- La direction d'un groupement
- Des "antennes" invisibles ayant les caractéristiques de l'élément duquel elles originent
- Les relations entre les éléments du groupement

2- Les principes:

- Tout groupement se constitue en traçant successivement des segments et des angles avec ces segments.
- Tout groupement se reconstitue en fonction des relations qui le définissent: ces relations proviennent de l'utilisation des commandes d'utilisation de code, d'échange et d'influence.
- Les relations qui construisent le groupement se déroulent de façon simultanée avec le déplacement de l'agencement (vitesse et direction).

- Tout groupement a une vitesse et une direction.
- Tout groupement qui touche un autre groupement, soit par ses antennes ou autrement, essaie d'entrer en relation avec le ou les éléments les plus près du lieu du contact.

3- Les processus:

- Tout groupement qui rencontre un autre groupement et qui peut entrer en relation par au moins un de ses éléments, constitue avec cet autre groupement un nouveau groupement.
- Tout groupement qui rencontre un autre groupement et qui ne peut voir ses éléments entrer en relation avec les éléments de l'autre groupement, pivote ou ricoche sur l'autre groupement en fonction de sa vitesse et de sa direction.
- Le choc de deux groupements modifie la vitesse et la direction des groupements.

4- Les cas limites:

- Tout groupement qui n'a plus de vitesse disparaît.

Maintenant que nous avons choisi le mode de représentation adéquat et les représentations des termes centraux, voyons dans quelle mesure ces choix sont valides.

4. Les représentations les unes par rapport aux autres

A partir des termes centraux, nous sommes amenés à concevoir le micromonde comme le résultat de la juxtaposition de quatre secteurs des sciences sociales. Ces secteurs sont ceux desquels originent les termes centraux. Nous avons maintenant établi, pour chacun de ces secteurs, les choix à la disposition de l'utilisateur, les éléments de représentation, les principes et les processus régissant ces représentations. Une dernière étape avant de passer à la réalisation du sous-système de spécification informatique consiste à valider ces choix en vérifiant le degré d'interférence des représentations les unes par rapport aux autres. Les résultats de cette vérification sont résumés dans le tableau 2.

Comportements graphiques / Structures-mères	CODE	ECHANGE	FORME	INFLUENCE
INFLUENCE:				
1. Déplacement ou non d'un élément attiré	Compatible, pas d'interférence	Compatible, pas d'interférence	Compatible, mais interférence: la relation influence peut modifier la forme	OK
2. Absorption d'un élément	Disparition de toutes les relations	→	→	→
3. Disparition d'un élément sans relation	Disparition de toutes les relations	→	→	→
ECHANGE:				
1. La taille stable	Compatible, mais interférence: la taille a une couleur, le segment discontinu a la couleur de la communication	OK	Compatible sans interférence	Compatible sans interférence
2. Grossissement et diminution de la taille	Idem	OK	Compatible mais interférence: la relation échange peut modifier l'allure de la forme	Compatible pas d'interférence
3. Un élément de taille 0 disparaît	Disparition de toutes les relations	→	→	→
4. Disparition d'un élément sans relation	Disparition de toutes les relations	→	→	→
CODE:				
1. Le clignotement	OK	Compatible, sans interférence	Compatible, sans interférence	Compatible, sans interférence
2. La coloration	OK	Idem	Idem	Idem
3. La coloration du segment se propage puis disparaît	OK	Compatible, mais interférence; la couleur en mouvement ≠ la couleur par défaut	→	Compatible, sans interférence
4. Disparition d'un élément sans relation	Disparition de toutes les relations	→	→	→
FORME:				
1. Ensemble de relations entre les éléments	Compatible, mais interférence: le segment discontinu prend la couleur de la communication	Compatible, sans interférence	OK	Compatible, mais interférence: les segments s'allongent ou rapetissent
2. Déplacement ou non des formes	Compatible, sans interférence	→	OK	→
3. Juxtaposition de deux formes	Compatible, mais interférence: si les codes sont compatibles, les segments et les éléments se fondent, sinon la relation de la forme attirante s'impose	→	OK	→
4. Disparition de la forme sans vitesse	Disparition de toutes les relations	→	OK	→

Tableau 2: Interférence des relations les unes par rapport aux autres

On voit, par ce tableau, que les choix de représentations que nous avons faits sont compatibles sans interférence ou compatibles avec interférence. Dans les cas d'interférence, on remarque enfin que cela ne remet pas en cause les choix que nous avons faits mais oblige à superposer les comportements en cause. Et cela s'avère possible dans tous les cas d'interférence.

III- LA REALISATION DU SOUS SYSTEME DE SPECIFICATION INFORMATIQUE

1. LE STRUCTUROGRAMME GENERAL

2. LES PAGES-ECRAN

   2.1 L'objet de transition

   2.2 Relation qui a un impact sur la taille

   2.3 Relation qui a un impact sur la couleur

   2.4 Relation qui a un impact sur la mobilité

   2.5 Relations qui agencent les éléments

   2.6 Caractéristiques concernant la présence de plusieurs relations sur un même segment et l'agencement d'éléments qui ont des interrelations

3. LES TABLES DE DECISION

   3.1 Association-échange

   3.2 Association-influence

   3.3 Association-code

   3.4 Association-forme

Troisième partie

La realisation du sous-système de spécification informatique

Les spécifications informatiques n'ajoutent rien aux différentes dimensions du langage du micromonde des sciences sociales contenues dans la seconde partie de ce rapport. Elles servent à détailler de façon technique les objectifs opérationnels du langage et, plus specifiquement, les éléments constituants et les processus des différentes représentations des termes centraux. Comme nous l'avons exposé en première partie, les spécifications informatiques comprennent un structurogramme général, des pages-écran de même que les tables de décision présentant l'ensemble des règles qui régissent les actions ces primitives à l'écran.

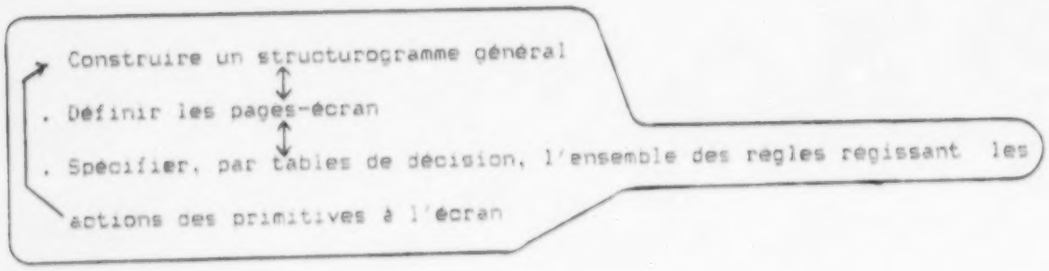

Construire un structurogramme général
. Définir les pages-écran
. Spécifier, par tables de décision, l'ensemble des règles régissant les actions des primitives à l'écran

Figure 8: Sous-système de spécification informatique

1. Le structurogramme général

Le structurogramme général présente une image de l'ensemble des composantes du logiciel. Nous pouvons y voir les quatre grandes fonctions du didacticiel:

1. Accéder au système d'aide
2. Réaliser un projet
3. Manipuler des fichiers
4. Se servir des utilitaires

Ces fonctions regroupent les objectifs opérationnels du langage. La prochaine figure présente le structurogramme général d'un micromonde.

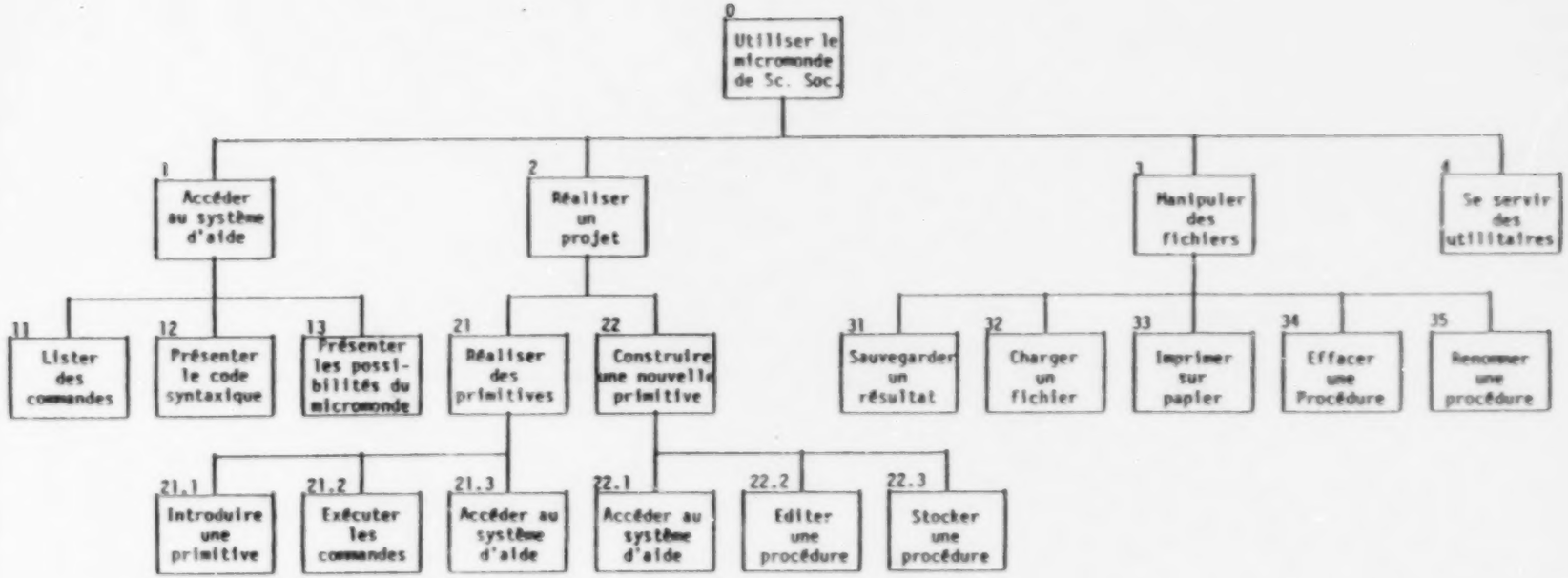


Figure 9: Structurogramme général

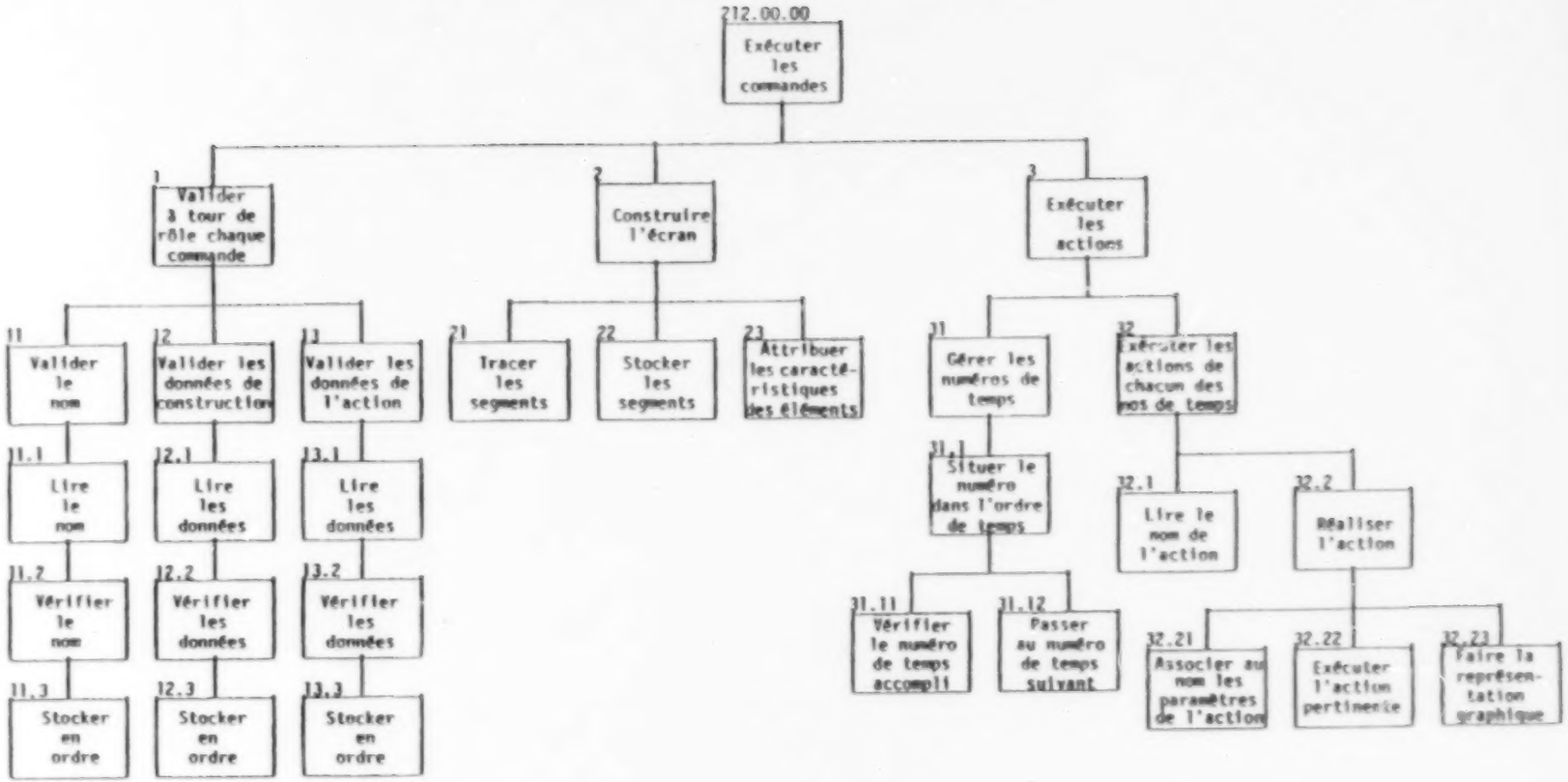


Figure 10: Aperçu du contenu de la section 212.00 du structurogramme général

Ce structurogramme ne présente pas de spécificité se rapportant aux sciences sociales. La case no 21.1 demande d'être précisée davantage pour connaître les primitives spécifiques aux sciences sociales et les représentations graphiques commandées. Les pages-écran offrent plus de précision.

2. Les pages-écran

Les pages-écran présentent, de façon statique, des dessins-types voulant illustrer l'action graphique commandée par chacune des primitives rattachées aux termes centraux des sciences sociales. Les caractéristiques que nous décrirons sont celles de l'objet de transition (2.1) et celles concernant les représentations graphiques utilisées pour concrétiser les structures-mères échange (modification de la taille des éléments 2.2), code (modification de la couleur des éléments 2.3), influence (modification de la localisation des éléments 2.4) et forme (modification de l'agencement des éléments 2.5).

A ces caractéristiques, il faut ajouter celles concernant l'utilisation simultanée des modifications de la taille, de la couleur et de la localisation des éléments (3.1) et enfin, celles concernant l'agencement des éléments en un groupement (3.2).

2.1 L'objet de transition

L'objet de transition est un point dans l'écran. Nous le nommerons "l'objet" et nous le représenterons par deux points (:) dans les figures qui suivent. Ce point est en fait un lieu (figure 11), une coordonnée à partir de laquelle on peut tracer un segment. Lorsqu'un segment est tracé, l'objet de transition voit son lieu modifié et se déplace au bout de ce segment (figure 12).

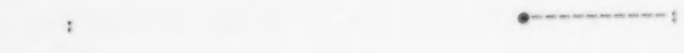


Figure 11

Figure 12

De ce lieu, il est possible de tracer des segments mais aussi de donner un angle, vers la gauche (G x) ou vers la droite (D x), à un segment à partir de l'orientation du dernier point du segment tracé (figure 13). Il est aussi possible de spécifier une longueur pour chaque segment à tracer (figure 14).

Figure 13

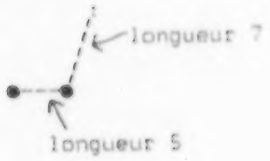


Figure 14

A partir de ce lieu, il est aussi possible de sauter à un endroit dans l'écran afin de modifier la coordonnée de l'objet (figure 15). L'angle servant de référence afin de calculer l'orientation d'un éventuel segment à partir de la nouvelle position de l'objet de transition est celui demandé pour effectuer le saut (figure 16).

saut

Figure 15

G45

Figure 16

L'utilisation de l'objet peut aussi se faire à l'aide de segments invisibles. Les résultats apparaissent à l'écran sous forme de

points (figure 17).

Figure 17

Chaque déplacement de l'objet exige donc une commande qui spécifie un angle et une longueur. Une suite de commandes donne donc une suite de segments (figure 18).

```
commande  G90,4
commande  G90,5
commande  D90,10
commande  D90,15
commande  D22,5
commande  D112,12
```

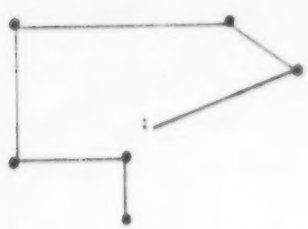


Figure 18

L'objet permet donc de situer des points dans l'écran et de les relier par des segments visibles ou invisibles. Ces segments servent à visualiser des liens qui sont en fait des relations. La nature de ces relations aura un impact sur la taille des points (relation d'échange), sur leur couleur (relation d'utilisation de codes), sur leur mobilité dans l'écran (relation d'influence) et sur leur disposition, leur agencement à l'écran (relation de forme).

2.2 Relation qui a un impact sur la taille

Une commande permet de spécifier un angle et une longueur à un segment. Elle permet donc de lier deux points dans l'écran. Ces points sont pour nous des éléments. Chaque élément peut être doté d'une certaine quantité qui détermine sa taille (figure 19 (x)). A l'aide de la même commande on peut spécifier divers flux (figure 19 (-->x/x<--)) entre ces éléments. Un élément donne donc à un autre qui peut donner lui aussi. Au total, selon qu'un élément reçoit plus qu'il ne donne, sa taille augmente (figure 20-a). S'il donne plus qu'il reçoit, sa taille diminue (figure 20-b). S'il donne autant qu'il reçoit, sa taille reste stable (figure 20-c). Seuls les éléments avec leur taille respective et les segments apparaissent à l'écran.

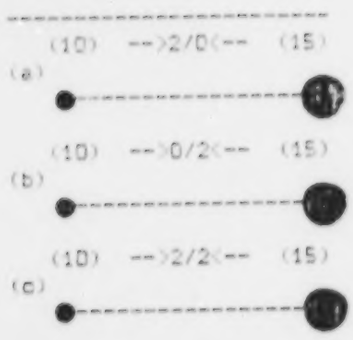


Figure 19

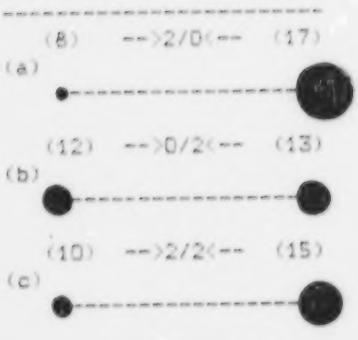


Figure 20

Dans la figure 19, nous avons trois relations qui illustrent les trois types d'échange possible. Ces relations pourraient résulter de trois commandes qui seraient alors des primitives et comporteraient les éléments suivants:

```
---------------------
nom de la commande, angle, longueur, dotation de
l'élément début,dotation de l'élément fin, flux de
l'élément début, flux de l'élément fin.
---------------------
```

ou

```
---------------------
commande d'échange 0,18,10,15,2,0
commande d'échange 0,18,10,15,0,2
commande d'échange 0,18,10,15,2,2
---------------------
```

Ces commandes utilisées telles quelles produiraient trois relations continues. Pour obtenir la disposition de la figure 19, il faudrait rédiger une procédure qui utiliserait la commande de saut de

l'objet afin que ce dernier saute, après avoir tracé la relation (a), à l'origine de la relation (b). La procédure se lirait alors comme suit:

```
----------------------
commande d'échange 0,18,10,15,2,0
commande de saut D135,22
commande d'échange G135,18,10,15,0,2
commande de saut D135,22
commande d'échange G135,18,10,15,2,2
----------------------
```

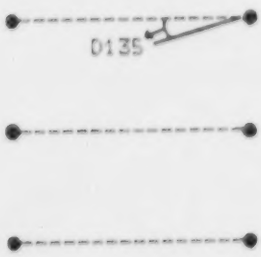
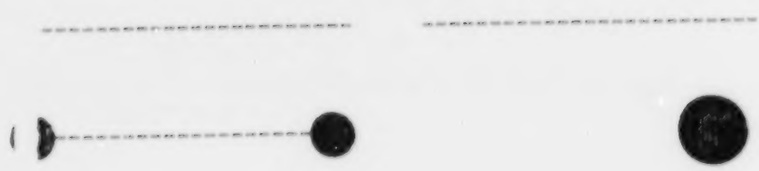


Figure 21

Ces relations ont aussi la particularité d'être dynamiques. Ceci signifie que le flux opère de façon constante jusqu'à la disparition d'un ou des éléments qui n'ont plus de dotation ou de taille. Ainsi la primitive: commande d'échange 0,18,10,10,10,0 produira, dans un premier temps, la relation illustrée à la figure 22. Compte tenu que le flux de l'élément début est égal à sa dotation, nous assisterons, dans un deuxième temps, à la disparition de l'élément début et à une multiplication par deux de la taille de l'élément fin (figure 23).



gure 22 Figure 23

Aussi, tout élément qui n'est pas en relation avec un autre ément disparaît de l'écran après un temps relativement court. nsi, l'élément restant à la figure 23 disparaîtrait lui aussi.

2.3 lation qui a un impact sur la couleur

La relation qui a un impact sur la couleur des éléments est une lation de communication. La composition des couleurs sert à créer s codes entre les éléments. Chaque élément a donc une couleur mais ssi un nombre de chances de communiquer.

La commande d'utilisation de code permet de mettre en relation ux éléments et de leur attribuer des chances de communiquer et une uleur-code. Deux éléments peuvent communiquer entre eux si la uleur de chacun d'eux est composée d'une couleur commune (figure). Une communication augmente d'une unité les chances de mmuniquer de l'élément qui la reçoit et diminue d'une unité les ances de communiquer de l'élément qui émet la communication (figure).

commande d'utilisation de code 0,17,2,4,bleu,vert

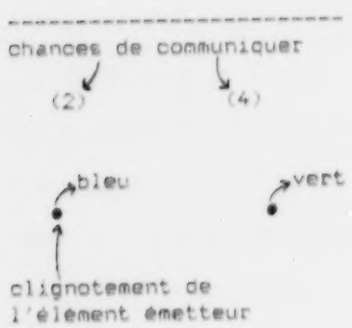


Figure 24

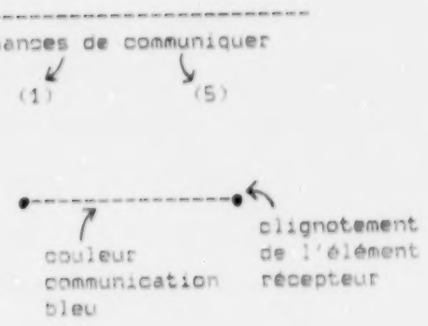


Figure 25

Quand les deux éléments clignotent, il s'établit un segment de l'élément émetteur vers l'élément récepteur, segment qui disparaît progressivement par l'arrière dès qu'il a atteint l'élément récepteur. Ce segment prend la couleur commune aux deux éléments.

La communication établie à l'aide du code est permanente. Ainsi, un élément qui est en relation de communication avec un autre élément, s'il ne reçoit pas de communication d'un autre élément, épuise ses chances de communiquer. Aussi, la composition des couleurs comme mode de création des codes a comme conséquences que les couleurs primaires ne peuvent pas permettre la communication sauf entre des éléments de la même couleur et que les couleurs tertiaires permettent un plus grand nombre de communications.

2.4 Relation qui a un impact sur la mobilité

La relation qui a un impact sur le déplacement des éléments est une relation d'influence. Chaque élément se voit attribuer, à l'aide de la commande d'influence, une valeur qui constitue sa force d'attraction ou son degré d'influence (figure 26). La force ou le degré d'influence de chaque élément fait en sorte que chaque élément peut agir sur un élément avec lequel il est mis en relation. Ainsi, un élément peut attirer vers lui un élément ayant une force d'attraction inférieure (figure 29). Si les forces d'attraction sont égales, les éléments en relation sont stables à l'écran (figures 26-27).

---

commande d'influence 0,17,12,12

Figure 26

Figure 27

commande d'influence 0,17,12,8

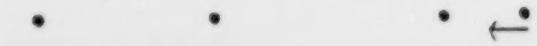


Figure 28

Figure 29

La vitesse avec laquelle un élément est attiré par un autre élément est fonction de la différence entre les forces d'attraction respectives de ceux-ci.

Aussi la force d'attraction est continue de telle sorte qu'un élément attiré par un autre disparaît au contact de ce dernier à moins qu'une autre force vienne contrer son déplacement.

2.5 Relations qui agencent les éléments

L'agencement des éléments se fait par le positionnement d'éléments reliés par des segments. Cette succession s'opère par une commande qui détermine un nombre de segments à tracer, un angle et une longueur (figure 29). Cet agencement est ensuite doté par la même commande d'une vitesse et d'une direction dans l'écran (figure

30). La direction se détermine par une première référence à l'un des quatre points cardinaux (nord=n, sud=s, est=e, ouest=o) suivi d'un nombre qui correspond à un endroit précis sur la longueur du côté choisi.

---

commande d'agencement 4,G90,10,10,N20

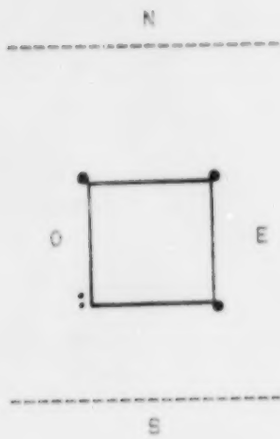


Figure 29

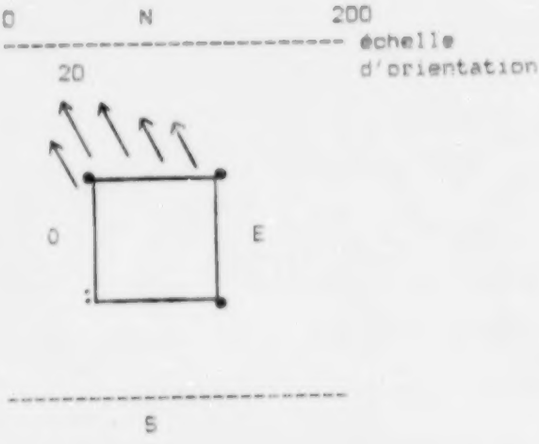


Figure 30

Ces agencements peuvent prendre différents aspects selon le nombre de segments, la longueur et l'angle demandé (figures 31-32).

commande d'agencement
3,G30,8,10,010

Figure 31

commande d'agencement
3,G150,3,5,010

Figure 32

La commande d'agencement permettra aussi d'augmenter pour chaque agencement sa capacité d'entrer en contact avec d'autres formes. Ainsi, lors de la construction de l'agencement, le logiciel devra construire et tenir compte d'"antennes" qui s'ajoutent mais sans que ces antennes apparaissent à l'écran. Leur longueur est fonction de l'angle demandé dans la commande d'agencement. Un angle qui tend à ouvrir l'agencement produit une "antenne" plus longue qu'un angle qui tend à fermer l'agencement sur lui-même (figures 32a-32b).

commande d'agencement
3,D135,8,10,S10

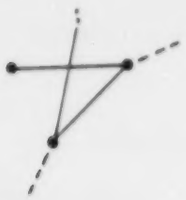


Figure 32a

commande d'agencement
3,D45,8,10,S10

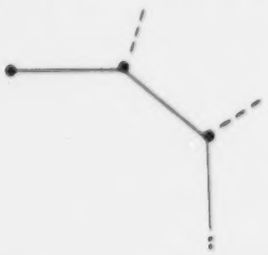


Figure 32b

Selon la vitesse et la direction de chaque agencement, il est possible que deux agencements se rencontrent. Il doit être prévu alors que ces agencements, selon les cas, roulent ou ricochent l'un sur l'autre et que leur vitesse et leur direction soient modifiées en conséquence (figures 33-34).

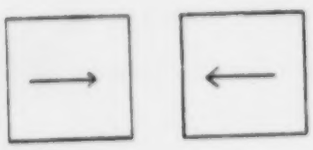


Figure 33

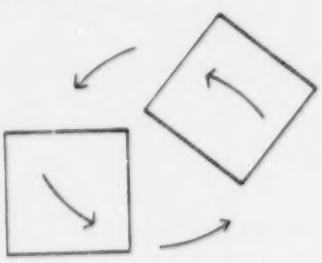


Figure 34

2.6. Caractéristiques concernant la présence de plusieurs relations sur un même segment et l'agencement d'éléments qui ont des interrelations

a) Concernant la présence de plusieurs relations sur un même segment

La conception du logiciel doit prévoir que les dimensions (modification de la taille des éléments, utilisation de la couleur, déplacement des éléments l'un par rapport à l'autre, agencement des éléments selon une forme donnée) programmées par l'utilisateur doivent se superposer à l'écran dans l'ordre où elles apparaissent dans la procédure. Le logiciel doit aussi tendre vers la simultanéité de la représentation des relations et des agencements programmés.

Ainsi une relation d'échange et d'influence doit être représentée simultanément par le déplacement des éléments s'il y a lieu et les modifications de taille nécessaires s'il y a lieu. Il en est de même pour la couleur représentant le code et la couleur du segment signifiant la communication.

Aussi tout élément qui coupe par son déplacement une relation déjà établie participe de cette relation s'il le peut, c'est-à-dire si l'utilisateur lui a attribué lors de la programmation des paramètres rendant possible d'entrer en relation avec les éléments de la relation coupée.

b) Concernant l'agencement d'éléments qui ont des interrelations (influence, code, échange)

L'agencement d'éléments doit pouvoir supporter que les segments de l'agencement soient en fait des relations d'utilisation de code et/ou d'échange et/ou d'influence. Un agencement à l'aide de ces relations s'appellera pour nous un groupement. Un groupement exigera, pour être construit, une commande particulière qui prendra le nom "groupement". Une telle commande introduira l'utilisateur dans une démarche particulière. Ainsi, la commande groupement suivi de la désignation d'une vitesse et d'une direction fait en sorte que les commandes qui suivent permettront la constitution d'un groupement.

Par exemple:

```
---------------------
commande de groupement 10,S20
commande d'échange D90,10,5,5,2,2
commande d'échange D90,10,10,10,3,3
commande d'influence D90,10,4,4
commande de code D90,10,10,10,bleu,vert
---------------------
```

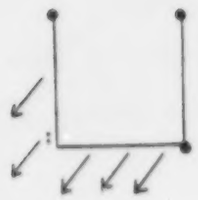


Figure 35

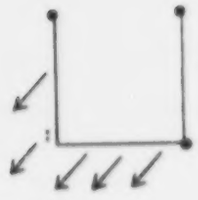


Figure 36

Il serait possible aussi d'utiliser une commande de retour de l'objet au début du dernier segment tracé. Ceci aurait pour effet de pouvoir multiplier les relations sur un même segment. L'exemple précédent deviendrait ainsi:

```
commande de groupement 10,S20
commande d'échange 90,10,5,5,2,2
commande d'échange ,début,10,10,3,3
commande d'influence ,début,4,4
commande de code ,début,10,10,bleu,vert
```

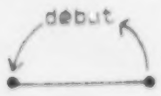


Figure 37

A l'aide des "antennes", la commande de groupement permettra aussi d'augmenter, comme la commande d'agencement, la capacité du groupement d'entrer en contact avec d'autres groupements.

Avec la commande de groupement, les "antennes" sont en fait constituées d'une serie de points qui ont les mêmes parametres que l'element d'où elles originent. Ainsi le logiciel, lorsque ces antennes entrent en contact avec un autre groupement (dans une relation ou sur un élement), vérifiera, à l'aide des parametres retenus pour chaque point de l'antenne (figure 38), si ces paramètres permettent que l'élement, ducuel originne l'antenne, entre en relation avec l'élément de l'autre groupement (figure 39) ou entre en relation avec deux elements déja en relation dans l'autre groupement (figure 40). Si aucune relation est possible, les groupements pivotent (figure 41) ou ricochent l'une sur l'autre (figure 42).

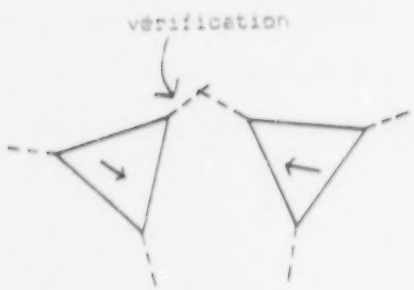


Figure 38

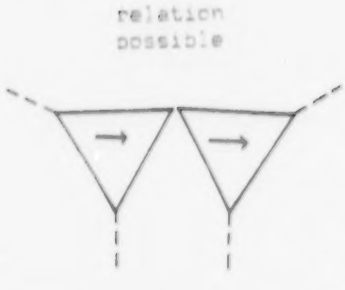


Figure 39

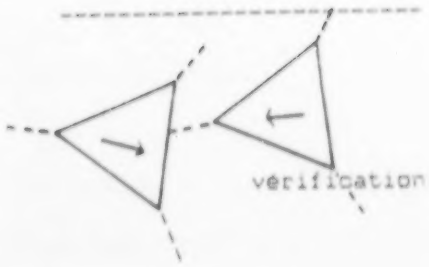


Figure 40

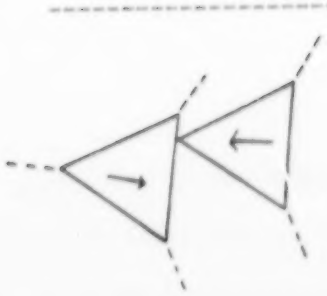


Figure 41

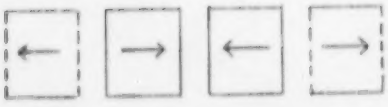


Figure 42

Il est à noter que les pages-écran ne peuvent pas représenter la réalité active de l'écran. Il faut les percevoir comme des images analytiques utiles pour faire compendre de façon littéraire ce qui sera surtout un dessin animé. La prochaine section précise davantage l'aspect dynamique de la représentation graphique.

3. Les tables de décision[6]

Les tables de décision sont des instruments techniques qui servent à faire le lien entre les promoteurs et les programmeurs. Elles servent à définir les conditions qui gèrent chacune des actions découlant des représentations graphiques des termes centraux. Voici les tables de décision correspondant à association-échange, association-influence, association-cons, association-forme et offrant aussi un guide des erreurs.

3.1 Association-échange

PRIMITIVE-ECHANGE (acceptable lexiquement et syntaxiquement)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Eléments en relation avec C	o	o	o	n	n	n	n	n	n	n	n	n	n	n	r	o	o	o	o	o	o	o	o	o	o	o	o	n	n	n
Rayon taille élément X) longueur du segment	n	n	n	o	o	o	n	n	n	n	n	n	o	o	o	n	n	n	o	o	o	o	o	o	o	o	o	n	n	n
Flux de A) taille A	n	n	n	n	n	n	o	o	o	n	n	n	o	o	o	o	o	o	n	n	n	o	o	o	o	o	o	n	n	n
Flux de B) taille B	n	n	n	o	o	o	n	n	n	o	o	o	o	o	o	o	o	o	o	o	o	n	n	n	o	o	o	n	n	n
Flux de A - Flux de B = 0	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n
Flux de A - Flux de B) 0	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n
Flux de A - Flux de B (0	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o	n	n	o
Tracer un segment et les éléments	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Augmenter la taille élément A en proportion			x		x			x			x			x			x			x			x			x			x
Diminuer la taille élément A en proportion				x					x											x									x	
Augmenter la taille élément B en proportion				x		x			x				x			x				x			x			x			x	
Diminuer la taille élément B en proportion					x			x																	x					x
Garder les tailles constantes			x		x			x			x			x			x			x			x			x			x	
Effacer élement A et segment vers B						x				x			x		x			x						x			x			
Effacer élément B et segment vers A						x				x			x		x			x			x						x			
Eloigner le plus petit élément			x		x									x						x										

Table de décision des erreurs de la primitive échange

Echange A jusqu'à TD TF ED EF

	R 1	2	3	4	5	6	7	8
Si TD = D < TD < TG	o	o	o	n	n	n	o	n
Si TF = D < TF < TG	o	o	n	o	n	n	o	n
Si E = D < X < TG	o	n	o	o	o	n	n	n

Module: erreur

X	Exécuter la commande
X X X X X X X	Envoyer un des messages d'erreurs suivants:
X X X	1. "Réévaluer le flux de l'échange"
X X X	2. "Réévaluer la taille de l'element F"
X X X	3. "Réévaluer la taille de l'element D"
X	4. "Verifier les arguments de la relation"

Symboles: TD = Taille de l'élément Debut
TF = Taille de l'élement Final
E = Flux de l'echange

3.2 Association-influence

INFLUENCE

	1	2	3	4	5	6	7	8
L = 0	n	n	n	n	o	o	o	o
IF = ID	o	n	n	n	o	n	n	n
IF > ID	n	o	n	n	n	o	n	n
IF < ID	n	n	o	n	n	n	o	r

(ne sont pas en relation)

x	x	x	Tracer un segment invisible					
x	x	x	Placer le point D				x	x	x
x	x	x	Placer le point F				x	x	x
x			Immobiliser D et F				x		
	x		Déplacer D vers F (L + (-1))						
		x	Déplacer F vers D (L + (-1))						
			Fusionner F avec D (L = D)					x	
		x	Faire disparaître F et D						x
			Fusionner D avec F (L = D)					x	

Symboles: ID = Force d'influence du point début
IF = Force d'influence du point final
L = Longueur du segment

Table de décision des erreurs de la primitive influence

Influence A jusqu'à L ID IF

	R_1	R_2	R_3	R_4
Si ID = 0 ≤ ID < TG	o	o	n	n
IF = 0 ≤ IF < TG	o	n	o	n

Module: erreur

X			Exécuter la commande
	X	X	Envoyer un message d'erreurs selon le cas
		X	X	1. "Vérifier la capacité d'influence de l'élément début"
	X		X	2. "Vérifier la capacité d'influence de l'élément final"

Symboles: ID = Force d'influence du point début
IF = Force d'influence du point final
TG = Trop grand

3.3 Association-code

#PRIMITIVE-CODE (acceptable syntaxiquement et logiquement)

	1 2 3 4	5 6 7 8	9 0 1 2 3 4	5 6	7 8 9 0	1 2 3 4	5 6 7 8 9 0	1 2
Couleur compatible	o o o o	o o o o	o o o o o o	o o	n n n n	n n n n	n n n n n n	n n
Chance de communication du receveur = 0	o n n n	n o o o	n o n o o n	o n	o n n n	n o o o	n o n o o n	o n
Chance de commun. de l'émetteur = 0	n o n n	o n o o	n o o n n o	o n	n o n n	o n o o	n o o n n o	o n
Communications reçues du receveur = chance de communication X 2	n n o n	o o n o	o n o n o n	o n	n n o n	o o n o	o n o n o n	o n
Communications reçues de l'émetteur = chance de communication X 2	n n n o	o o o n	o n n o n o	o n	n n n o	o o o n	o n n o n o	o n
Augmenter nombre de communication du receveur	x x x x	x x x x	x x x x x x	x x				
Diminuer nombre de communication de l'émetteur	x x x x	x x x x	x x x x x x	x x				
Augmenter chance de communication du receveur	x x x x	x x x x	x x x x x x	x x				
Diminuer chance de communication de l'émetteur	x x x x	x x x x	x x x x x x	x x				
Faire clignoter l'émetteur	x x x x	x x x x	x x x x x x	x x	x x x x	x x x x	x x x x x x	x x
Faire clignoter le receveur	x x x x	x x x x	x x x x x x	x x				
Tracer le trait mobile	x x x x	x x x x	x x x x x x	x x				
Faire disparaître l'émetteur	x	x	x x	x x	x	x		
Faire disparaître le receveur		x	x x	x	x	x	x	

Table de décision des erreurs de la primitive code

Communique A jusqu'à L ND, NF, CD, CF

	R_1	2	3	4	5	6	7	8	9	1_0	1_1	1_2	1_3	1_4	1_5	1_6
Si CD=0< CD<16	o	o	n	n	n	o	o	o	n	n	n	o	o	o	n	n
Si CF=0< CF<16	o	n	o	n	n	o	n	n	o	n	o	o	o	n	o	n
Si ND=0< ND<TG	o	n	n	o	n	n	o	n	n	o	o	o	n	o	o	n
Si NF=0< NF<TG	o	n	n	n	o	n	n	o	o	o	n	n	o	o	o	n
	X															Exécuter la commande
		X	X	X	X	X	X									Envoyer un message selon le cas
		X	X	X	X										X	1. "Vérifier les arguments de la relation
						X		X				X				2. "Réévaluer la capacité de communication de l'élément D"
						X	X	X			X	X				3. "Réévaluer la capacité de communication de l'élément F"
							X		X	X					X	4. "Vérifier le code de communication de l'élément D"
								X		X	X				X	5. "Vérifier le code de communication de l'élément F"

Module: Erreur

Symboles: CD = Couleur de l'élément Début
CF = Couleur de l'élément Final
ND = Nombre de communication allouée à l'élément Début
NF = Nombre de communication allouée à l'élément Final
TG = Trop grand

3.4 Association-forme

PRIMITIVE-FORME (acceptable syntaxiquement, lexiquement et logiquement)

	1 2 3 4 5	6 7 8 9 0	1 1 1 1 1	1 2 3 4 5	1 1 1 1 2	6 7 8 9 0	2 2 2 2 2	1 2 3 4 5	2 2 2 2 3	6 7 8 9 0	3 3 3 3 3	1 2 3 4 5	3 3 3 4 4	6 7 8 9 0
Vitesse de A < vitesse de B	o o o o o	n n n n n	n n n n n	o o o o o	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n
Vitesse de B < vitesse de A	n n n n n	o o o o o	n n n n n	n n n n n	n n n n n	o o o o o	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n
Vitesse de A = vitesse de B	n n n n n	n n n n n	o o o o o	n n n n n	n n n n n	n n n n n	o o o o o	n n n n n	n n n n n	n n n n n	o o o o o	n n n n n	n n n n n	n n n n n
Vitesse de A ou B = 0	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	o o o o o	o o o o o	o o o o o	o o o o o
La relation est possible	o o o o o	o o o o o	o o o o o	n n n n n	n n n n n	n n n n n	n n n n n	n n n n n	o o o o o	o o o o o	o o o o o	o o o o o	o o o o o	o o o o o
Une antenne touche une relation	o n n n n	o n n n n	o n n n n	o n n n n	o n n n n	o n n n n	o n n n n	o n n n n	o n n n n	o n n n n	o n n n n	o n n n n	o n n n n	o n n n n
Une antenne touche un élément	n o n n n	n o n n n	n o n n n	n o n n n	n o n n n	n o n n n	n o n n n	n o n n n	n o n n n	n o n n n	n o n n n	n o n n n	n o n n n	n o n n n
Un élément touche une relation	n n o n n	n n o n n	n n o n n	n n o n n	n n o n n	n n o n n	n n o n n	n n o n n	n n o n n	n n o n n	n n o n n	n n o n n	n n o n n	n n o n n
Un élément touche un élément	n n n o n	n n n o n	n n n o n	n n n o n	n n n n o	n n n o n	n n n o n	n n n o n	n n n o n	n n n o n	n n n o n	n n n o n	n n n o n	n n n o n
Les antennes et les éléments ne se touchent pas	n n n n o	n n n n o	n n n n o	n n n n o	n n n n o	n n n n o	n n n n o	n n n n o	n n n n o	n n n n o	n n n n o	n n n n o	n n n n o	n n n n o
La ou les formes apparaissent pendant un certain temps et disparaissent							x x x x x	x x x x x						
Les formes se déplacent en fonction de leur direction et de leur vitesse			x		x		x		x		x		x	
Les formes se juxtaposent au niveau du point de jonction	x x x x		x x x x		x x x x									
Une nouvelle vitesse et une nouvelle direction sont acquises	x x x x		x x x x		x x x x									
La forme A pivote sur la forme B à partir du point de jonction								x x x x				x x x x		
La forme B pivote sur la forme A									x x x x		x x x x			
La vitesse de A est modifiée								x x x x		x x x x		x x x x		
La vitesse de B est modifiée								x x x x		x x x x		x x x x		
La direction de A est modifiée								x x x x		x x x x		x x x x		
La direction de B est modifiée								x x x x		x x x x		x x x x		

Table de décision des erreurs de la primitive groupe

Groupe X A jusqu'à L, ... , V, D

	R_1	2	3	4	
Si V = TP < V < TG	o	o	n	n	Module: erreur
Si D = 0 < D < 360°	o	n	o	n	
	X				Exécuter la commande
		X	X	X	Envoyer un message d'erreur selon le cas:
	X			X	1. "Votre groupement n'a pas d'orientation"
			X	X	2. "Votre groupement n'a pas de mouvement

Sympoles: V = Vitesse
D = Orientation
X = Nombre de segments
TP = Trop petit
TG = Trop grand

TABLE DE DECISION

Fonction contrôle

(Association) A jusqu'à L

	R_1	2	3	4
Si A = 0 ≤ A < 360°	o	o	n	n
L = dans l'écran	o	n	o	n

Module: Erreur

X			Prendre le reste de la commande
	X	X	X	Envoyer un message d'erreur selon le cas
		X	X	1. "Verifier l'angle de la relation"
	X		X	2. "Verifier la longueur de la relation"

Symboles A = Angle
L = Longueur du segment

Conclusion

Notre objectif pour l'année 84/85 était de réaliser la conception informatique d'un micromonde de sciences sociales à partir des résultats obtenus à l'étape de la conceptualisation. Plus précisément, il s'agissait de se donner une démarche et des instruments qui nous permettent de décrire à des programmeurs les termes centraux transformés en primitives d'un langage informatique spécifique à un micromonde de sciences sociales. Cet objectif a été atteint tant du point de vue du contenu (transformation des termes centraux en primitives) que du point de vue des instruments (réalisation du sous-système de spécification informatique). En effet, nos instruments de présentation des primitives se sont avérés suffisants dans chacune de nos communications avec nos consultants en informatique.

Nous avions planifié une analyse organique des représentations et des primitives comme une suite logique de la construction des tables de décision et du structurogramme. Une analyse de faisabilité du micromonde (voir annexe) nous a révélé que les infrastructures québécoises de la production informatique, tant privée que publique, présentent une expertise assez limitée dans le domaine de la conception de langage de programmation.

Les délais et les coûts de réalisation d'une analyse organique

satisfaisante débordaient largement les cadres de notre projet. Aussi, avons-nous décidé de rendre encore plus explicites les primitives et leurs représentations. Nos contacts avec divers consultants et des analystes-programmeurs nous ont permis de constater que ce choix était judicieux.

Sur le plan des résultats globaux, il nous apparaît que le micromonde de sciences sociales est suffisamment élaboré pour passer à l'étape de la réalisation informatique. Aussi, il constitue pour nous le terme de deux années de recherche qui démontrent que le concept de micromonde (dont la tortue-logo est une application) peut être appliqué à des domaines comme ceux des sciences humaines. A notre point de vue, cela est d'autant plus intéressant et important, que ces domaines d'application sont des domaines quelque peu négligés par les applications pédagogiques de l'ordinateur.

Références

1. Checkland, P.B., Systems Thinking, Systems Practice, Wiley, 1981.

2. Fournier C., Lafeuille J., Conceptualisation d'un micromonde en sciences sociales, Cégep de Chicoutimi, 1984, p.37.

3. Fournier C., Lafeuille J., op.cit. p.32.

4. Papert, Seymour, Le jaillissement de l'esprit, Flammarion, Paris, 1981, p.149.

5. Fournier C., Lafeuille J., Conceptualisation d'un micromonde en sciences sociales, Cégep de Chicoutimi, 1984.

6. Frohock, Fred M., "The Structure of Politics", The American Political Science Review, vol. 72, Iss. no 3, Sept. 1978, p.859-869.

7. Fournier C., Lafeuille J., Conceptualisation d'un micromonde en sciences sociales, Rapport de l'enquête Delphi, Annexe, Cégep de Chicoutimi, 1984.

8. Belbahri, Nadir, Informatique de base, Inforum Inc., 1981, page 333.

ANNEXE

MLG
Maziade, Laroche, Gagnon
CONSEILLERS EN INFORMATIQUE

DOCUMENT D'ORIENTATION SUR LE PROJET DE MICROMONDE EN SCIENCES SOCIALES

Chicoutimi
le 27 janvier 1985

Siège social 930, Jacques-Cartier est, Suite D-415, Chicoutimi, Qc Adresse postale C.P. 880, Chicoutimi, Qc G7H 5E8 - tél.: (418) 543-0666

INTRODUCTION

Notre bureau d'études spécialisé en recherch et innovation, analyse fonctionnelle et études d'implantation, fut contacté par les auteurs, Madame Claire Fournier et Mon ieur Jacques Lafeuille, au sujet de leur projet de Micromond en Sciences sociales.

Il s'est dégagé des rencontres et discus ions préliminaires avec ceux-ci, un mandat pour la firme MLC Consultants en Informatique de développement, d'une lig e à suivre quant à l'orientation que pourrait prendre le projet.

A cet effet, une première partie du document con erne la réalisation technique du projet de Micromonde et ensuite, nous passons à la structure de l'organisation qui peut le mettr en oeuvre.

Il est bien entendu impossible ici de faire autre hose que d'esquisser une vue d'ensemble du projet de façon à écl irer au mieux les auteurs et ceux qui les ont chargés de ce tr vail sur l'ampleur du projet présenté pour fin d'étude.

PARTIE 1

En premier lieu, nous allons définir les contraintes techniques qui nous guideront par la suite dans les choix à prendre dans la réalisation du problème:

- Contrainte graphique.

  La réalisation du modèle visuel présuppose une assez bonne définition au niveau de l'écran.

- Contrainte de rapidité.

  Afin de mettre en relief les notions "d'échange" et "d'influence", le modèle visuel doit être animé d'une dynamique acceptable.

- Contrainte de facilité.

  L'apprentissage du système par l'étudiant doit être assez facile et complémentaire au niveau du menu d'aide, de façon à ne pas décourager l'utilisateur peu familier avec ce type d'instrument.

- Contrainte de globalité.

  Le modèle doit pouvoir si besoin est, se décomposer en partie, mais tout en tenant compte et ce, en représentant la notion de contrainte de facilité, de la synthèse c'est à-dire de la vue d'ensemble du modèle.

Après avoir identifié ces contraintes, et tout en n'oubliant pas que ce système reposera sur une machine, reportons-les maintenant en termes techniques:

- Contrainte graphique.

Pour avoir une bonne définition visuelle des modèles développés, c'est-à-dire un cercle qui ressemble à un cercle et ainsi de suite, et ce faisant, faire appel à la notion très importante dans ce cas-ci de la couleur, l'écran doit être subdivisé en pixels graphiques adressables au besoin il va s'en dire. La définition minimale serait de 128,000 pixels (640 X 200) ou à l'optimum de 256,000 pixels (512 X 512).

- Contrainte de rapidité.

C'est ici que se situe le point le plus sensible du système analysé et voici pourquoi:

Le système doit:

1. Interpréter les commandes de l'usager.
   - Effectuer la prise de données.
   - Rechercher dans la table de relations.
   - Bâtir le modèle ou le modifier.

2. Effectuer les calculs nécessaires à la mise en forme de l'écran.

3. Rafraichir l'écran en temps réel et revenir en 1.

Sur la plupart des micro-ordinateurs susceptibles de fonctionner en milieu scolaire, le niveau de rapidité d'exécution d'une tâche est relativement lent.

C'est pourquoi, la réalisation du logiciel sera effectuée presque entièrement en langage machine, de façon à gérer le système entièrement et probablement même, par dessus le système d'exploitation, qui sera pour les besoins de la cause, annihilé par le système.

Bien sûr, avec l'évolution de la quincaillerie, il sera possible et même essentiel de programmer en langage évolué tel le C ou ADA, ce dernier, faisant appel à des concepts abstraits très puissants qui permettront d'alléger de beaucoup la programmation

du système, tout en gardant des routines de base (I/O) en langage machine.

- Contraintes de rapidité.

Afin d'utiliser au maximum l'élément "facilité", une souris serait dans ce cas-ci hautement appréciée et ce, en addition avec un menu d'aide à l'écran du genre LOTUS 1-2-3 ou SYMPHONY.

Il est très important, étant donné que l'usager dans ce cas-ci sera vraisemblablement quelqu'un qui n'aura pas une grande expérience sur le plan informatique, que la clarté prime avant toute chose.

- Contrainte de globalité.

En terme technique, s'il est possible de faire un zoom particulaire au niveau du modèle, il devra toujours être possible de faire un retour en arrière pour avoir une vue d'ensemble du système soit en ouvrant une fenêtre à l'écran (window) ou du moins au minimum, en y accédant très facilement.

Toutes ces contraintes au niveau technique, nous donnent déjà les premiers paramètres du système:

- Possibilité d'une bonne capacité de mémoire RAM (256 K par exemple).
- Gestion graphique par pixels.
- Processus assez rapide au niveau des I/O et des calculs.
- Interface si possible avec un langage évolué et puissant tel le ADA ou, dans une moindre mesure le C afin d'améliorer la transportabilité des programmes source, en vue d'une utilisation sur plusieurs types d'équipements.

PARTIE 2

La deuxième partie de ce présent document concerne la réalisation proprement dite du logiciel, c'est à-dire un estimé de réalisation lié aux contraintes définies ans la première partie.

Ici, deux choix s'offrent à nous, l premier, lié au modèle LOGO, le deuxième, au modèle ZERO.

Le modèle LOGO:

Il ne faut pas perdre de vue ici, qu les auteurs dans leur démarche de Micromonde, ont adopté beaucou de concepts liés au modèle LOGO et que par le fait même, des co tacts pourraient être tentés en vue d'une coopération bénéfique ux deux parties.

Le modèle ZERO:

Celui-ci, beaucoup plus complexe, pa t du postulat que tout est à faire, à partir de zéro:

- Etude de marché, réseau de distribution, e c...
- Analyse fonctionnelle.
- Test et prototype.
- Finalisation et correction d'analyse.
- Programmation et test.

CONCLUSION

Nous avons agi dans ce dossier à titre de consultant et d'orienteur, et nous croyons avoir rempli à bien ce mandat avec la rencontre des auteurs et les gens de LOGO en janvier 85.

Nous croyons que la solution première, c'est-à-dire le modèle LOGO, est le meilleur et le plus susceptible de répondre aux exigences du problème, tout en assurant aux auteurs une chance que leur modèle soit appliqué dans un délai raisonnable.

En effet, la similitude des deux démarches ne s'explique que par la complémentarité de l'une par rapport à l'autre.

Nous croyons que le Ministère de l'Education du Québec aurait des réticences à embarquer dans un tel projet vu l'ampleur de celui-ci, toutefois, une solution mitoyenne telle que proposée ici, serait sûrement acceptable pour les deux parties, en ce sens qu'elle ne reléguerait pas aux oubliettes une étude qui, nous croyons, apporte un élément de fraîcheur et un créneau bien particulier et souvent délaissé de l'usage des ordinateurs dans le domaine des Sciences Humaines.

Ici s'arrête notre démarche pour l'instant dans ce dossier.

Nous sommes bien entendu disponibles pour tout nouveau mandat sur ce dossier, et nous publierons sous peu le résultat de tests techniques effectués sur des équipements divers en vue d'une compréhension plus poussée du problème posé.